National Taiwan Ocean University

Department of Computer Science & Engineering

MingleNet:
A Novel Dual Stacking Approach for Medical Image Segmentation

Jiunn wei Chiang 姜竣維 **00957056@mail.ntou.edu.tw**
Bao Wei Chiam 詹堡崴 **00957058@mail.ntou.edu.tw**
Dwayne Reinaldy 許漢強 **00957059@mail.ntou.edu.tw**
Vincensius Hobart Wijaya 何天勝 **00957060@mail.ntou.edu.tw**
Hanson Gabriel Cendana 曾漢盛 **00957062@mail.ntou.edu.tw**

Advisor： Dr. Kuan Y. Chang 張光遠

2023 / 12 / 01

Explanation of Project Roles and Contributions

Number	Name	Primary Job Responsibilities	Contribution to the Project (100%)
1	姜竣維	Poster Creation, Task allocation, Record Video	15%
2	詹堡崴	Report writing, Presentation, Record video, Base model compare	20%
3	許漢強	Report writing, Literature Review, Experiment implementation, Base Model Compare	25%
4	曾漢盛	Report writing, Literature review, Experiment implementation, Base Model Compare	25%
5	何天勝	Literature review, Diagram Design, Figure Design	15%

Table of Contents

Abstract

Medical image segmentation is important for disease diagnosis and treatment planning. Ensemble learning, which combines multiple models or predictions, can improve accuracy and performance in medical image segmentation. We propose MingleNet, which uses multiple layers of ensemble learning. MingleNet uses double-stacking of models, such as DoubleU-Net, DeepLabv3+, U-Net, and DeepLab, to produce masks. The first layer's masks are averaged and concatenated with the original images for the second layer. We also apply dynamic data augmentation to enhance model performances. We evaluate MingleNet on polyp segmentation benchmark datasets: Kvasir-SEG, CVC-ClinicDB, and CVC-ColonDB. On Kvasir-SEG, MingleNet achieves 93.19% Dice, 87.24% IoU, 94.15% precision, 92.25% recall, and 97.87% accuracy. On CVC-ClinicDB, MingleNet achieves 95.99% Dice, 92.29% IoU, 96.08% precision, 95.90% recall, and 99.21% accuracy. On CVC-ColonDB, MingleNet achieves 94.33% Dice, 89.28% IoU, 95.89% precision, 92.83% recall, and 99.10% accuracy. Our proposed method demonstrated competitive performance across the Kvasir-SEG, CVC-ClinicDB, and CVC-ColonDB datasets. On the CVC-ClinicDB and CVC-ColonDB benchmarks, MingleNet ranks **1st** in Dice and IoU. Moreover, MingleNet ranks **8th** in Dice and **17th** in IoU on the Kvasir-SEG benchmark.

Introduction

Medical image segmentation serves the critical purpose of enhancing the visibility of anatomical structures within images, which enables more precise analysis and diagnosis [1]. By partitioning the image into different regions, it allows for extraction of relevant features and improves diagnostic accuracy. With the advancements in imaging technologies, like magnetic resonance imaging (MRI) scans and computed tomography (CT) scans, medical image segmentation has garnered significant interest and attention from researchers and practitioners alike in recent years.

The goal of medical image segmentation is to accurately describe the boundaries and contours of organs, tissues, or other structures of interest within medical

images. The purpose of this process is to extract quantitative information, volumetric measurements, and spatial relationships, which are vital for clinical decision-making and precise intervention.

A wide range of segmentation techniques have been developed and employed in the field of medical imaging. Traditional methods include thresholding, region-based methods, clustering, and edge detection [1,2,3]. Nevertheless, these approaches often struggle with the complexity and variability of medical images and it leads to non-optimal results.

Medical image segmentation has been revolutionized in recent years by the emergence of deep learning-based techniques, with convolutional neural networks (CNNs) playing a pivotal role in this transformation. CNNs have shown exceptional performance in various segmentation tasks by automatically learning hierarchical representations and capturing contextual information from medical images. Models such as U-Net[4], SegNet[5], DeepLab[6], TransUNet [7], UNet++[8], and V-net[9] have been extensively used and adapted for medical image segmentation. U-Net[4], SegNet[5], DeepLab[6], UNet++[8], DeepLabv3+[10], and DoubleU-Net[11] are all deep CNN models for semantic segmentation, featuring an encoder (compression) path to capture context and a decoder (expansion) path for precise localization. From the encoder to the decoder, U-Net[4] transfers the entire feature maps, while SegNet[5] transfers only the pooling indices. U-Net uses skip connections to fuse low-level and high-level features, and up-convolutions to upsample the feature maps[4]; DeepLab uses atrous convolutions to enlarge the receptive field of the feature maps, and atrous spatial pyramid pooling to capture multi-scale information[6]. UNet++ is an improved version of U-Net, which uses nested and dense skip connections to enhance the feature fusion between the encoder and the decoder.[8] Instead, TransUNet is a transformer-based neural network that combines the strengths of both transformers and U-Net [7]. V-Net is a 3D CNN designed for volumetric image segmentation, known for its effectiveness in tasks like organ and tumor segmentation in medical imaging [9]. DeepLabv3+ is an advanced iteration of DeepLab featuring spatial attention mechanisms to enhance focus on relevant regions and improve localization[10]. DoubleU-Net is an extension of the original

U-Net architectures, designed to further enhance segmentation performance by incorporating dual encoding and decoding pathways to capture more intricate features[11].

Ensemble learning is a machine learning technique that integrates the predictions of multiple models to improve a single model by reducing variance and exploiting the advantages of each model [12]. One of the most popular methods is Stacking. Stacking trains multiple models (usually of different types) on the same data and then uses another model, which is called a meta-learner, to learn how to best combine their predictions [13].

In this study, we explored whether integrating ensemble learning techniques could substantially improve the performance of medical image segmentation models. To choose our base model, we carefully reviewed the existing literature, focusing on benchmark results such as Kvasir-SEG[14], CVC-ClinicDB [15], and CVC-ColonDB[16] in paperswithcode benchmark[17]. We opted for DeepLabv3+[10] and DoubleU-Net[11] specifically because both have high performance in segmentation tasks and both models are the best models derived from two different models, U-Net[4] and DeepLab[6]. Therefore, we have thought about using both uniqueness to our advantage by using ensemble learning to increase performance. By using those models we leverage the diverse architectures and unique strengths of each model by capturing complimentary patterns and robustness on diverse datasets.

Motivation & Purpose

The main challenge in medical image segmentation is the varying performance of models across different datasets. This variability is often due to differences in data characteristics and model architecture. Precise prediction is crucial in this field, making it important to understand and address this variability.

Our study is motivated by the critical need to confront the inherent variability in model predictions. This variability is not merely an inconvenience, but a barrier to achieving consistent and reliable results across diverse datasets, particularly when

models are constructed using different architectures. The subtle complexities within these models can exert a profound influence on their performance. We aim to unravel this issue and determine whether ensemble learning techniques can emerge as a robust remedy. Ensembles have the capability to combine multiple models and harness the collective wisdom of diverse architectures, holding the potential to mitigate the nuanced peculiarities that compromise prediction stability.

Related Works

DivergentNets

DivergentNets is a medical image segmentation technique that uses an ensemble of multiple high-performing image segmentation architectures [18]. The model combines the TriUNet segmentation model with an ensemble of well-known segmentation models, namely UNet++, FPN, DeepLabv3, and DeepLabv3+ [18]. The TriUNet model takes a single image as input and then passes it into two distinct U-Net models [18]. The outputs of these two models are combined and then processed through a third U-Net model to generate the final segmentation masks [18].

In the EndoCV2021 challenge, the TriUNet architecture used in DivergentNets was the winning model in terms of segmentation accuracy and generalization [18]. In contrast to the original TriUNet method, DivergentNets employs an ensemble of five intermediate models (TriUnet, UNet++, FPN, DeepLabv3, and DeepLabv3+), which are trained separately and then combined by averaging the pixels between each mask[18].

In comparison, DivergentNets employs five different medical image segmentation models, while MingleNet uses four models, DeepLabv3+[10], DoubleU-Net[11], DeepLab[6], and U-Net[4]. Furthermore, MingleNet employs a stacking approach with multiple layers of models, utilizing two layers of deep learning models instead of the single layer used in DivergentNets. We employ the identical method, specifically the averaging technique to combine predictions.

Materials and Methods

Polyp Image Datasets

1. **Kvasir-SEG[14]**

   The Kvasir-SEG v2 dataset which consists of 1000 gastrointestinal segmented polyp images and corresponding segmentation masks, was manually annotated and verified by medical experts. The image resolution ranges from 332 x 487 to 1920 x 1072 pixels.

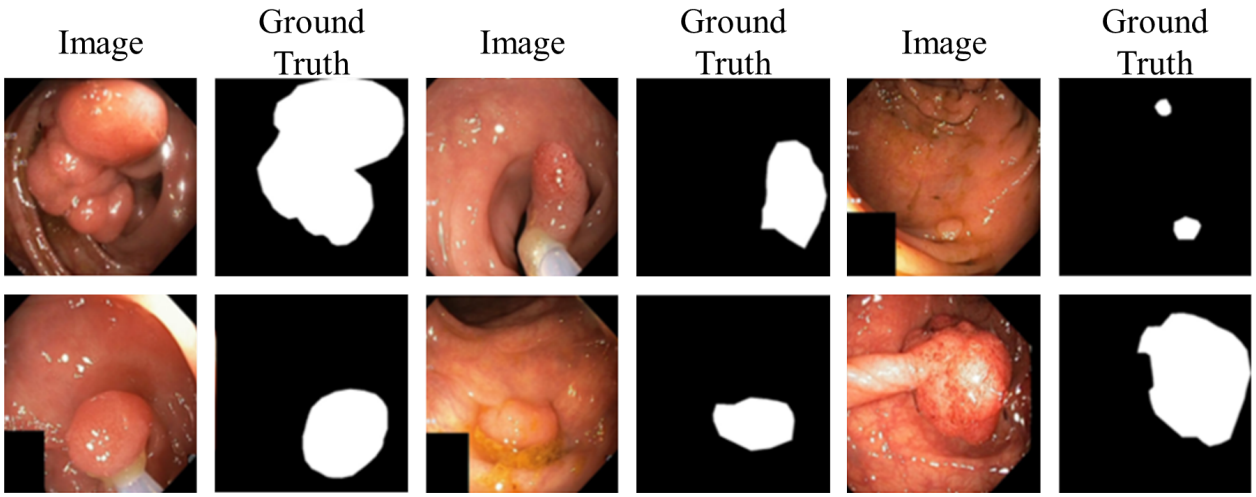

Fig. 1. Example of Images and Masks from Kvasir-SEG.
Each of the six images has a mask on its right side.

2. **CVC-ClinicDB[15]**

   CVC-ClinicDB is an open-access dataset of 612 gastrointestinal segmented polyp images and their ground truth from 31 colonoscopy videos at a resolution of 384x288 pixels.

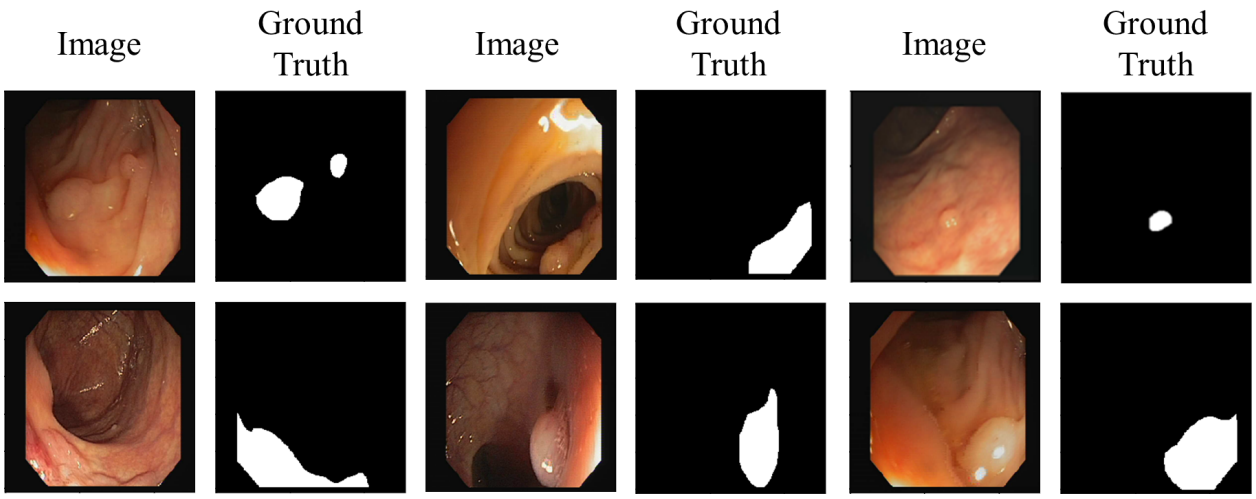

Fig. 2. Example of Images and Masks from CVC-ClinicDB.
Each of the six images has a mask on its right side.

3. **CVC-ColonDB[16]**

   CVC-ColonDB is an open-access dataset of 380 colon-segmented polyp images and masks from 13 patients' colonoscopy videos at 500 x 574 pixels.

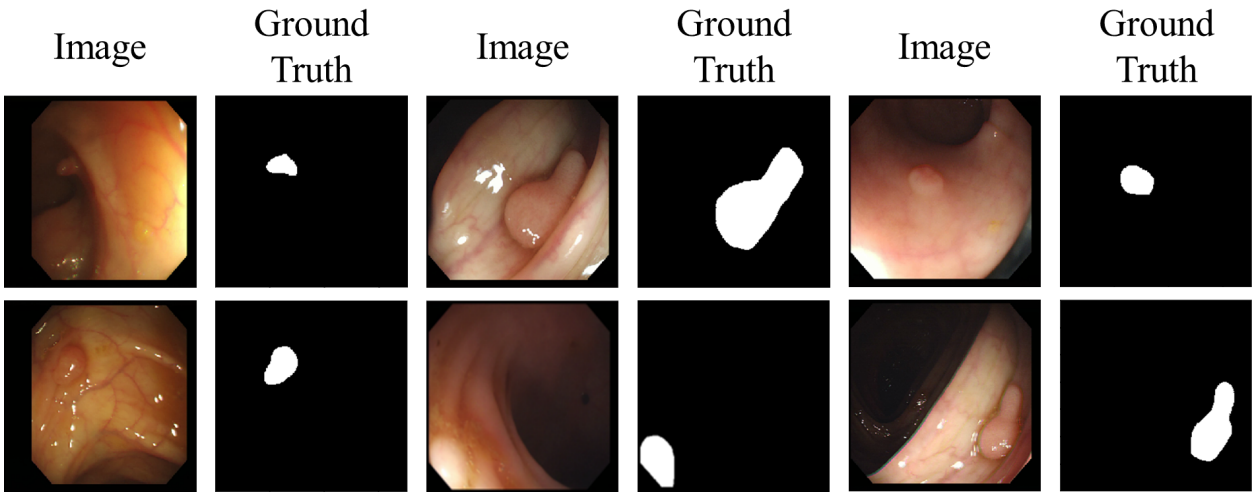

Fig. 3. Example of Images and Masks from CVC-ColonDB.
Each of the six images has a mask on its right side.

In this study, we rescaled Kvasir-SEG and CVC-ColonDB images to 352 x 352 pixels and CVC-ClinicDB images to 256 x 256 pixels.

Deep Learning Models for Image Segmentation

1. **DeepLabv3+**

   DeepLab is a semantic segmentation architecture that utilizes dilated convolutions to process the input image. The output is then fine-tuned by passing it through the fully connected CRF and bilinear interpolation. DeepLabv3+ incorporates semantic information from its encoder module and recovers precise object boundaries through the decoder module. DeepLabv3+ is enriched with semantic information from the encoder module, while the decoder module effectively recovers detailed object boundaries. The encoder module allows for the extraction of features at any resolution by using atrous convolution. The output of DeepLabv3 encodes ample semantic information, while atrous convolution offers control over the density of encoder features.

   The decoder module facilitates the precise recovery of object boundaries. DeepLabv3+ also explores this operation and shows improvement in terms of both speed and accuracy by adapting the ResNet101 model [10].

2. **DoubleU-Net**

   DoubleU-Net is a semantic segmentation model consisting of two U-Net architectures. It contains five components: two U-Net networks, VGG19, a squeeze-and-excite block, and atrous spatial pyramid pooling (ASPP). The first U-Net utilizes VGG19 as an encoder and ASPP as the input of the decoder. The squeeze-and-excite block improves the feature maps in the encoder of the first U-Net and those of both decoders. The output of the first U-Net and the corresponding image's mask are multiplied by the input pictures. The second U-Net also utilizes ASPP and the squeeze-and-excite block. Finally, the result is obtained through concatenation of the outputs from both U-Nets. DoubleU-Net strengthens the segmentation process by employing two U-Nets [11].

3. Comparison between DeepLabv3+ and DoubleU-Net

DeepLabv3+[10] emphasizes the use of atrous convolutions for capturing multiscale contextual information. It is renowned for its exceptional performance in segmentation tasks, particularly in situations where contextual information is essential.

DoubleU-Net[11] expands upon the U-Net framework by incorporating a dual pathway for information exchange. Its primary objective is to fuse features across multiple scales, thereby enhancing its ability to handle objects of various sizes.

DoubleU-Net[11] and DeepLabv3+[10] both are utilizing ASPP (Atrous Spatial Pyramid Pooling) as part of their architecture. Moreover, these models are also utilizing backbone in their architecture (DoubleU-Net[11] using VGG19 as a backbone, while DeepLabv3+[10] using ResNet101 as a backbone).

Concept of Ensemble Learning

Ensemble learning is a powerful machine learning approach that involves combining multiple individual models to achieve more accurate predictions or classifications than what a single model can achieve on its own. The fundamental idea behind ensemble learning is to leverage the diversity of these models to overcome the limitations and biases inherent in individual models, thereby enhancing overall performance and generalization capabilities[13].

Stacking

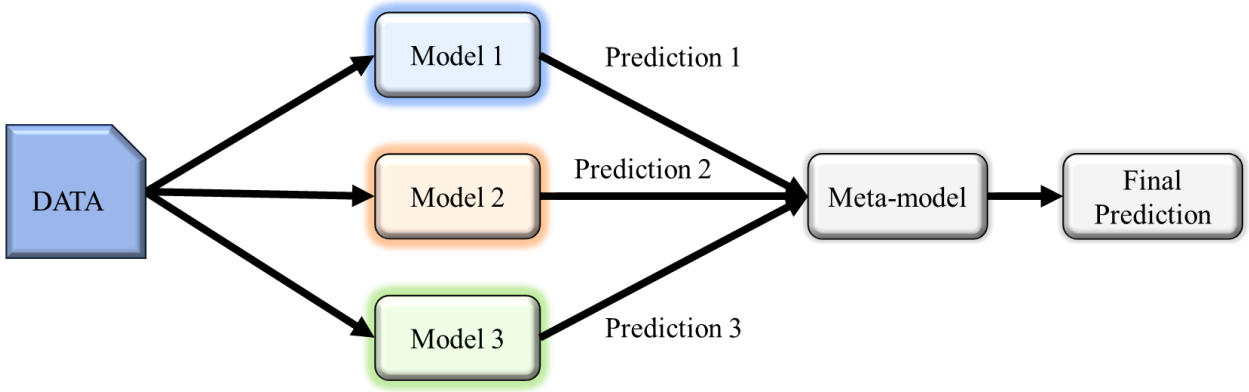

Fig. 4. Concept of Stacking

Stacking is an ensemble learning technique that improves predictive performance by combining multiple models. It uses their predictions as inputs to a meta-model, which makes the final prediction. The meta-model compensates for the weaknesses of individual base models, leading to better performance than using a single base model [13].

MingleNet Architecture

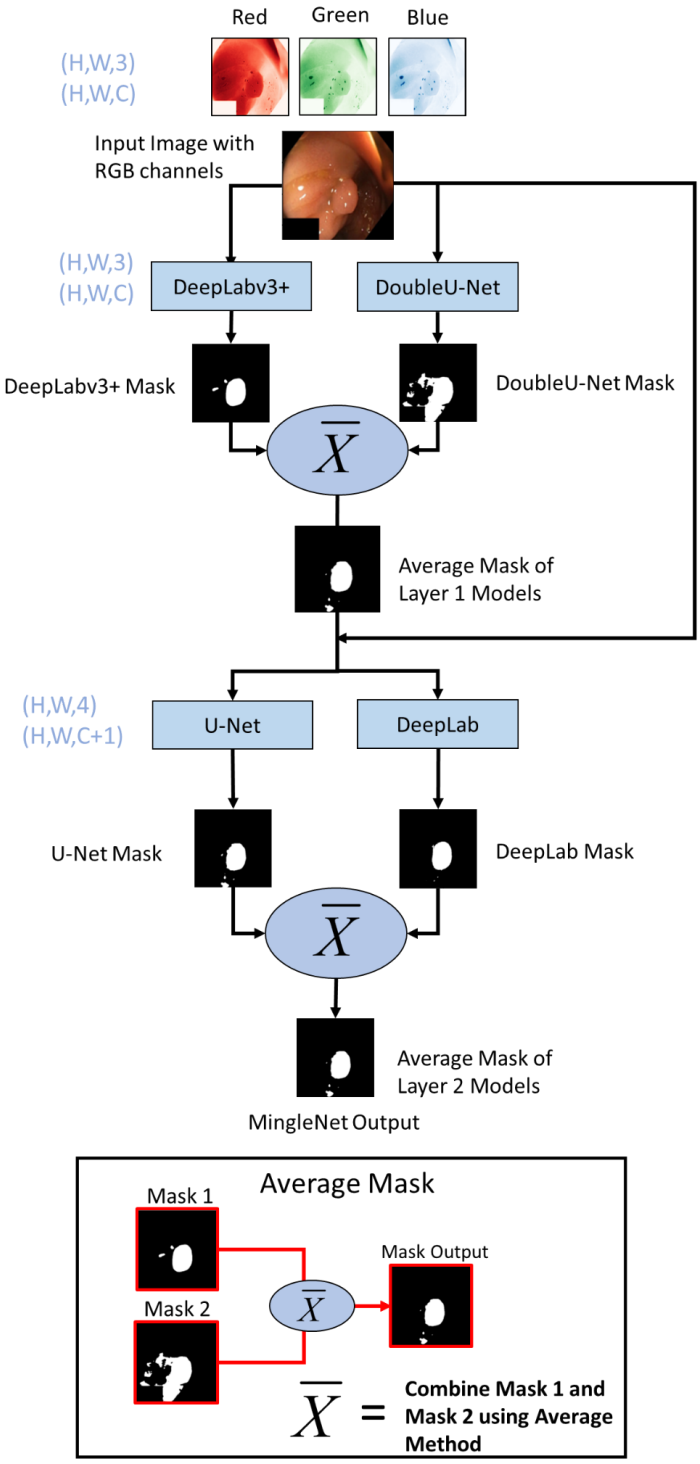

Fig. 5. MingleNet architecture

Diagram of the proposed ensemble learning technique

MingleNet is a convolutional neural network (CNN) architecture that fuses the incorporation of multiple layers of models, this architecture is illustrated in the accompanying Fig.3. In the architecture of MingleNet, the first layer consists of two different deep learning models, namely DeepLabv3+[10] and DoubleU-Net[11]. The first layer model takes the original image with three channels, Red, Green, and Blue(RGB) as input. Subsequently, we average the output of the first layer which consists of three different output masks. After averaging, the average mask output of the first layer is concatenated with the channels of the original input image, this resulted in the image consisting of four different channels (Red, Green, Blue, and average output masks of the first layer).

The second layer consists of two distinct deep learning models namely U-Net[4] and DeepLab[6]. The second layer model takes the image of four different channels (Red, Green, Blue, and average output masks of the first layer) as an input. Afterward, we average the output masks of the second layer as the final output. In addition, we selected U-Net[4] and DeepLab[6] for the second layer instead of DeepLabv3+[10] and DoubleU-Net[11] because DeepLabv3+[10] employs a ResNet101 backbone, and DoubleU-Net[11] utilizes a VGG19 backbone, both of which lack support for a 4-channel input.

We conducted training for 600 epochs on each model in the first layer, employing a batch size of 4, and saved the best-performing model. Furthermore, we opted for AdamW Optimizer with a learning rate of 0.0001, weight_decay of 0.004, and binary_crossentropy loss as the objective function for the first layer. Additionally, for the first layer, we conducted dynamic data augmentation for every epoch.

For the second layer, we conducted training for 200 epochs on each model, employing a batch size of 4. In addition, we chose the AdamW Optimizer and binary_crossentropy loss as the objective function, concurrently lowering the learning rate to 0.00001 with a weight_decay of 0.004. This adjustment aims to minimize the likelihood of the models bypassing optimal parameters, facilitating a more effective settling into the minimum of the loss function. Furthermore, in the second layer, dynamic data augmentation was omitted due to the time complexity

associated with its implementation. The dynamic data augmentation process necessitates loading, predicting, and averaging first-layer models for each epoch before concatenating them with the RGB input for the second layer.

Implementation Details

We used an Intel Core i7-12700 processor with an RTX 3080 TI GPU and 32GB of RAM at the Computational Biology Lab at the National Taiwan Ocean University to run Tensorflow 2.14.0 and Python 3.10.13.

Dynamic Data Augmentation

We used *Dynamic Data Augmentation*, a modified version of Sanderson and Matuszewski's method [19], to generate new augmented images before each training epoch. This technique improved the model's generalization. We applied the augmentation to the Albumentations library [20].

The Augmentation techniques parameter that we used is inspired by DUCK-Net [21] with the addition of Gaussian blur and adding the probability of 0.5 to horizontal flips, vertical flips, and Gaussian blur:

1. Horizontal flips and vertical flips (with a probability of 0.5),
2. Color jitter is applied with a brightness range of [0.6, 1.6], a fixed contrast of 0.2, a saturation factor of 0.1, and a hue factor of 0.01,
3. Affine transformation with rotations within the range of [-180°, 180°], horizontal and vertical translations within [-0.125, 0.125], scaling with a magnitude within [0.5, 1.5], and shearing with an angle within [-22.5°, 22°].
4. Gaussian blur with blur limit of [25,25], sigma limit of [0.001, 2.0], and probability of 0.5.

The reason why we apply this dynamic data augmentation is for adding variation to datasets. By applying flip, brightness, rotation, blur, etc it can improve the datasets variability. These transformations contribute to making our models more flexible and adaptable, as they learn to handle a broader range of scenarios.

Moreover, the reason why we apply probability to horizontal and vertical flips is to add even more data variation for each epoch so the model can generalize better. This allows each epoch to have only horizontally flipped images or only vertically flipped images or have both horizontal and vertical flip or no flip at all. As for color jitter and affine transformation, this data augmentation has a specific range of values. Therefore, each epoch has different color jitter and affine transformations as these parameters are randomly chosen within the ranges.

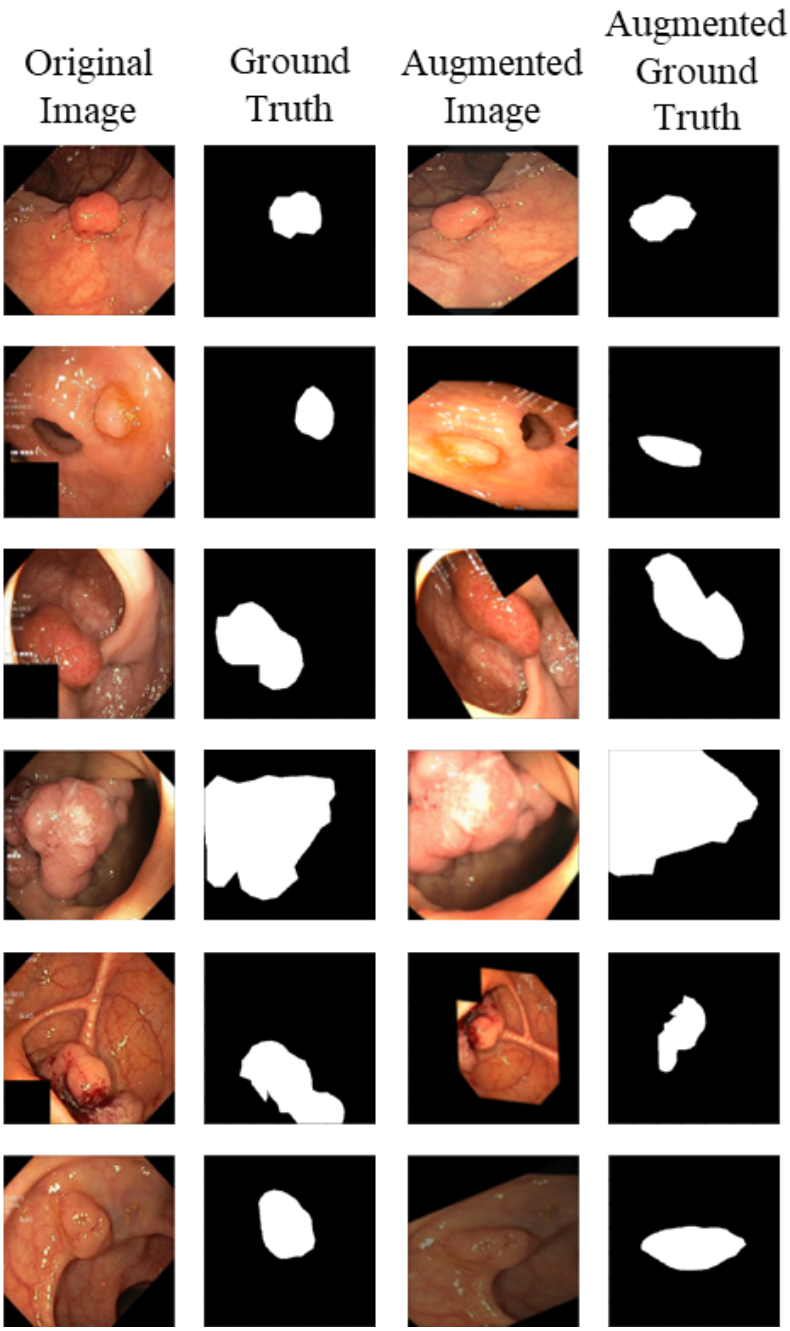

Fig. 6. Example of images, masks, augmented images, and augmented masks. Each of the six images is followed by a mask, an augment image, and its corresponding augmented mask.

Evaluation

For evaluation, we use the following performance metrics: Dice-Coefficient, IoU (Intersection over Union), Precision, Recall, and Accuracy. All these metrics are calculated regarding the true mask that we possess. The formula is presented below:

$$DiceCoefficient = \frac{2TP}{2\,TP+FP+FN}$$

$$IoU = \frac{TP}{TP+FP+FN}$$

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

All the metrics utilized in this study rely on the binary classification's confusion matrix, where TP, FP, TN, and FN correspond to the true positive, false positive, true negative, and false negative rates, respectively [22]. This approach ensures that our evaluation process is grounded in the comparison between the predicted results and the ground truth, represented by the true mask.

In this study, we have chosen not to employ Receiver Operating Characteristic (ROC) and Area Under the Curve (AUC) as the primary metrics for evaluating our image segmentation model. The rationale behind this decision stems from the nature of our specific task, which involves delineating objects and regions of interest within images with the availability of true masks (ground truth). ROC and AUC metrics are typically well-suited for assessing the trade-offs between sensitivity and specificity when making decisions based on class probabilities or scores. In contrast, image segmentation focuses on pixel-level delineation, and our primary objective is to measure the spatial overlap and accuracy of our model's predictions with respect to the ground truth. Due to this reason, metrics like Dice, IoU, Precision, Recall, and Accuracy are better aligned with our objectives, providing a more meaningful and detailed assessment of the quality and accuracy of our segmentation results. These metrics offer insights into how effectively our

model captures the spatial distribution of objects within images, making them the preferred choice for our evaluation.

Results

Table 1 shows that dynamic data augmentation improves all semantic segmentation models significantly. We use Dice-Coefficient and IoU as evaluation metrics, because they are more important for image segmentation. The most improved model is DDANet[26], which increases Dice by 14.82% and IoU by 20.68% after applying dynamic data augmentation. DeepLabv3+[10] is already a good model for the Kvasir-SEG dataset, but it still improves by 5% after applying dynamic data augmentation. This experiment demonstrates that dynamic data augmentation enhances the performance of each deep learning model.

Table 1. Model performance with and without Dynamic Data Augmentation on the Kvasir-SEG benchmark

Kvasir-SEG				
Model	Without Dynamic Data Augmentation		With Dynamic Data Augmentation	
	Dice	IoU	Dice	IoU
SegNet[5]	70.25%	55.52%	77.39%	63.12%
DDANet[26]	71.66%	55.84%	86.48%	76.18%
DeepLab[6]	75.41%	66.21%	87.61%	77.95%
U-Net[4]	83.59%	74.28%	86.73%	76.57%
DoubleU-Net[11]	83.62%	71.86%	92.58%	86.18%
DeepLabv3+[10]	87.20%	77.30%	91.86%	84.94%

Table 2 presents the performance of various MingleNet components on the Kvasir-SEG[14] benchmark. DoubleU-Net[11] outperforms DeepLabv3+[10] in Dice, IoU, and Recall at the first layer. At the second layer, U-Net[4] performs better than DeepLab[6] in Dice, IoU, and Recall after averaging the predicted masks from DoubleU-Net[11] and DeepLabv3+[10]. Comparing DoubleU-Net[11]

at the first layer, MingleNet's final output improves Dice by 0.61% and IoU by 1.13%.

Table 2. Performance of MingleNet on the Kvasir-SEG benchmark.

Kvasir-SEG					
Model	Dice	IoU	Precision	Recall	Accuracy
DoubleU-Net (Layer 1)	92.58%	86.18%	92.28%	**92.87%**	97.64%
DeepLabv3+ (Layer 1)	91.86%	84.94%	93.76%	90.03%	97.48%
U-Net (Layer 2)	93.13%	87.14%	93.92%	92.35%	97.84%
Deeplab (Layer 2)	92.69%	86.37%	**95.12%**	90.37%	97.74%
MingleNet's Final Output	**93.19%**	**87.24%**	94.15%	92.25%	**97.87%**

The best results are in **bold**.

Table 3 compares the performance of various semantic segmentation models on the Kvasir-SEG benchmark. Meta-Polyp[31] achieves the highest scores in Dice (95.90%) and IoU (87.24%). DUCK-Net[21] achieves the highest scores in Precision (96.28%), Recall (93.79%), and Accuracy (98.42%). These results show the effectiveness of the proposed MingleNet architecture in semantic segmentation tasks. Although MingleNet does not outperform models like DUCK-Net[21] and Meta-Polyp[31], it still shows robust performance for this dataset.

Figure 7 shows the progressive result in MingleNet on CVC-ClinicDB test set. Starting from base models consisting of DoubleU-Net[11] and DeepLabv3+[10], whose predictions are averaged and fed pass into the second layer comprising U-Net[4] and DeepLab[6]. Then MingleNet Output is the result of the averaged prediction masks of U-Net[4] and DeepLab[6].

Figure 7 reveals that the resulting output shows that MingleNet were to identify the shape of the polyp images. It also shows fewer false positives compared to the base models predictions and were able to identify the shape of the polyp objects, however results in the first row shows that DeepLab[6] prediction results have

shown a better visualization showing less false positives compared to MingleNet Output.

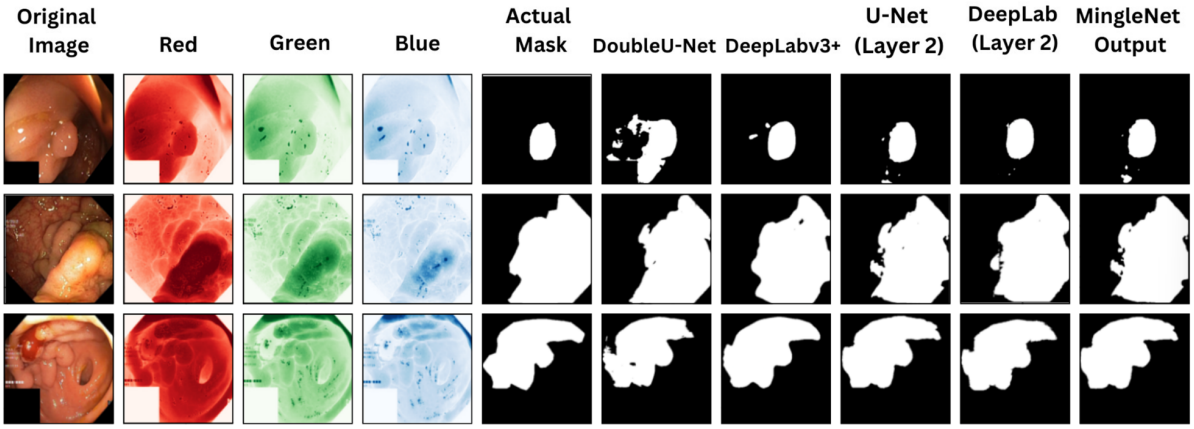

Fig. 7. Segmented polyp masks of MingleNet components on Kvasir-SEG benchmark.

The figure displays the actual mask, RGB channel, and the result of each segmentation from the first layer (DoubleU-Net and DeepLabv3+), the second layer (U-Net, DeepLab), and the final output.

Table 3. Segmentation Accuracy of our model on Kvasir-SEG Dataset compared to other models.

Kvasir-SEG					
Model	Dice	IoU	Precision	Recall	Accuracy
U-Net [4]	83.59%	74.28%	86.45%	76.14%	93.94%
U-Net (with our augmentation)[4]	86.73%	76.57%	90.04%	83.65%	95.95%
MSRF-Net (pre-trained)[23]	85.08%	74.04%	89.93%	80.74%	95.43%
HRNetV2 (pre-trained)[24, 25]	85.30%	74.38%	87.78%	82.97%	95.39%
DDANet(pre-trained)[26]	85.76%	78.00%	88.80%	86.43%	-
TransResU-Net (pre-trained)[27]	88.84%	82.14%	91.06%	90.22%	96.51%
PraNet(pre-trained)[28]	90.94%	83.39%	95.99%	86.40%	97.38%
FCN-Transformer (pre-trained)[29]	92.20%	85.54%	92.38%	92.03%	97.49%
DUCK-Net (pre-trained)[21]	95.02%	90.51%	**96.28%**	**93.79%**	**98.42%**
Meta-Polyp (pre-trained)[31]	**95.90%**	**92.10%**	-	-	-
MingleNet	93.19%	87.24%	94.15%	92.25%	97.87%

The best results are in **bold**.

Table 4 presents the performance of various MingleNet components on the CVC-ClinicDB[15] benchmark. DoubleU-Net[11] outperforms DeepLabv3+[10] in Dice and IoU at the first layer. At the second layer, U-Net[4] performs better than DeepLab[6] in Dice, IoU, and Precision after averaging the predicted masks

from DoubleU-Net[11] and DeepLabv3+[10]. Comparing DoubleU-Net[11] at the first layer, MingleNet's final output improves Dice by 0.10% and IoU by 0.19%.

Figure 8 shows the progressive result in MingleNet on CVC-ClinicDB test set. Starting from base models consisting of DoubleU-Net[11] and DeepLabv3+[10], whose predictions are averaged and fed pass into the second layer comprising U-Net[4] and DeepLab[6]. Then MingleNet Output is the result of the averaged prediction masks of U-Net[4] and DeepLab[6]. Figure 8 reveals that the resulting output shows that MingleNet were to identify the shape of the polyp objects. All rows show that it successfully reduced false positives that the base models predicted, improving segmentation.

Table 4. Performance of MingleNet on the CVC-ClinicDB benchmark.

CVC-ClinicDB					
Model	Dice	IoU	Precision	Recall	Accuracy
DoubleU-Net (Layer 1)	95.89%	92.11%	95.61%	**96.17%**	99.18%
DeepLabv3+ (Layer 1)	95.19%	90.83%	95.72%	94.67%	99.05%
U-Net (Layer 2)	95.94%	92.20%	**96.17%**	95.71%	99.20%
DeepLab (Layer 2)	95.56%	91.51%	94.69%	96.46%	99.11%
MingleNet's Final Output	**95.99%**	**92.29%**	96.08%	95.90%	**99.21%**

The best results are in **bold**.

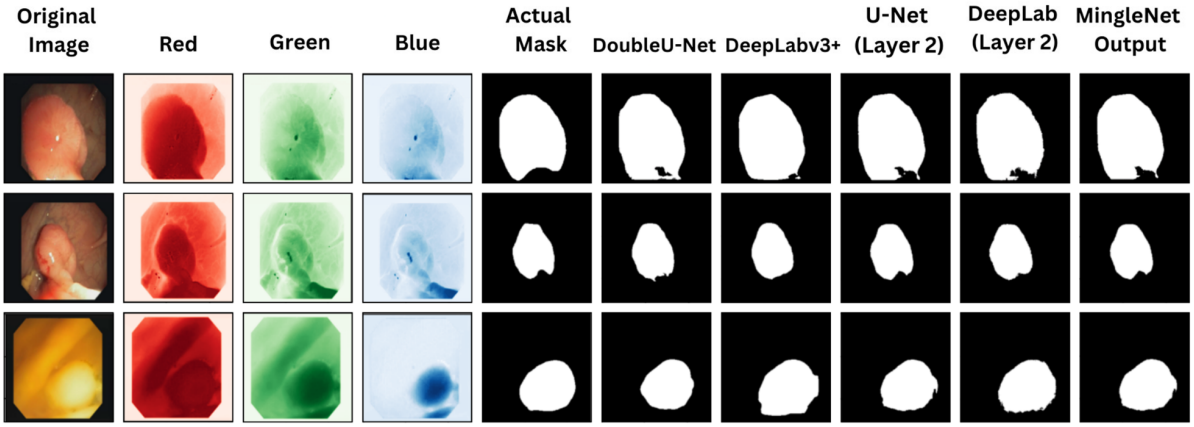

Fig. 8. Segmented polyp masks of MingleNet components on the CVC-ClinicDB benchmark.

The figure displays the actual mask, RGB channel, and the result of each segmentation from the first layer (DoubleU-Net and DeepLabv3+), the second layer (U-Net, DeepLab), and the final output.

Table 5 presents a comparative evaluation of various semantic segmentation models. Standing out among the evaluated models is our model MingleNet, which attains the highest scores in Dice(95.99%), Iou(92.29%), Recall(95.95%), and Accuracy(99.21%), except for Precision in which FCN-Transformer achieve the highest with a score of 96.59%. These results underscore the efficacy of the proposed MingleNet architecture in semantic segmentation tasks, outperforming other segmentation models in Dice and IoU.

Table 5. Segmentation Accuracy of our model on CVC-ClinicDB compared to other models.

CVC-ClinicDB					
Model	Dice	IoU	Precision	Recall	Accuracy
U-Net (without our augmentation)[4]	91.06%	81.51%	95.26%	87.22%	97.77%
U-Net (with our augmentation)[4]	91.42%	84.20%	93.29%	89.63%	98.34%
MSRF-Net (pre-trained)[23]	51.52%	34.69%	39.39%	74.43%	77.42%
HRNetV2 (pre-trained)[24, 25]	55.31%	38.22%	42.42%	79.44%	79.31%
PraNet(pre-trained)[28]	68.52%	52.12%	76.47%	62.07%	91.30%
FCN-Transformer (pre-trained)[29]	88.00%	78.58%	**96.59%**	80.82%	96.45%
DUCK-Net (pre-trained)[21]	94.78%	90.09%	94.68%	94.89%	99.07%
MingleNet	**95.99%**	**92.29%**	96.08%	**95.90%**	**99.21%**

The best results are in **bold**.

Table 6 presents the performance of various MingleNet components on the CVC-ColonDB[16] benchmark. DoubleU-Net[11] outperforms DeepLabv3+[10] in Dice, IoU, and Recall at the first layer. At the second layer, DeepLab[6] performs better than U-Net[4] in Dice, IoU, and Recall after averaging the predicted masks from DoubleU-Net[11] and DeepLabv3+[10]. Comparing DoubleU-Net[11] at the first layer, MingleNet's final output improves Dice by 0.63% and IoU by 1.21%.

Figure 9 shows the progressive result in MingleNet on CVC-ColonDB test set. Starting from base models consisting of DoubleU-Net[11] and DeepLabv3+[10], whose predictions are averaged and fed pass into the second layer comprising U-Net[4] and DeepLab[6]. Then MingleNet Output is the result of the averaged prediction masks of U-Net[4] and DeepLab[6]. Figure 9 reveals that the resulting output shows that MingleNet was able to identify the shape of the polyp objects, although there are some issues in Row 3 where it successfully decreased its false negatives in the polyp object area but still showed false positives in the outer parts.

Table 6. Performance of MingleNet on the CVC-ColonDB benchmark.

CVC-ColonDB					
Model	Dice	IoU	Precision	Recall	Accuracy
DoubleU-Net (Layer 1)	93.74%	88.21%	94.94%	92.56%	99.00%
DeepLabv3+ (Layer 1)	90.37%	82.44%	**96.29%**	85.14%	98.53%
U-Net (Layer 2)	93.60%	87.97%	95.88%	91.42%	98.99%
DeepLab (Layer 2)	93.91%	88.52%	94.74%	**93.08%**	99.02%
MingleNet's Final Output	**94.33%**	**89.28%**	95.89%	92.83%	**99.10%**

The best results are in **bold.**

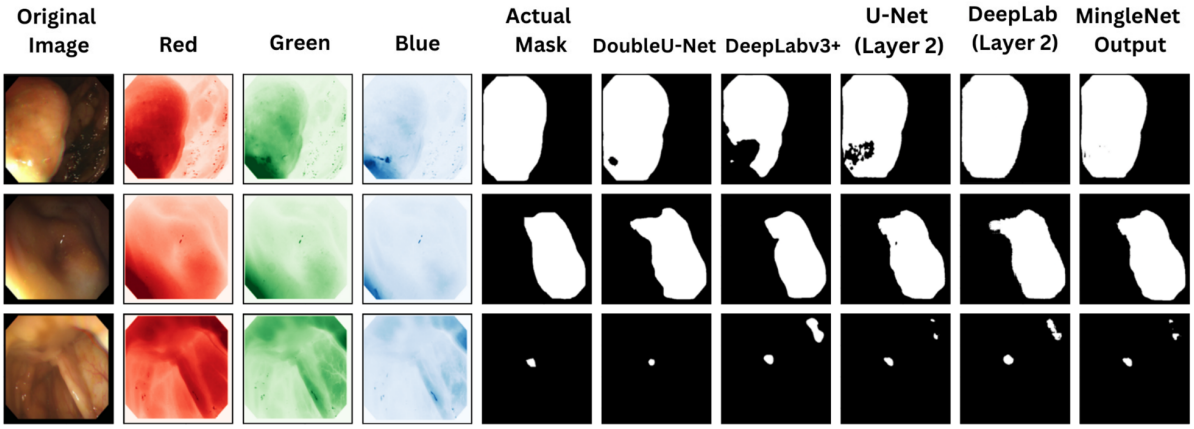

Fig. 9. Segmented polyp masks of MingleNet components on CVC-ColonDB benchmark.

The figure displays the actual mask, RGB channel, and the result of each segmentation from the first layer (DoubleU-Net and DeepLabv3+), the second layer (U-Net, DeepLab), and the final output.

Table 7 presents a comparative evaluation of various semantic segmentation models. Standing out among the evaluated models is our model MingleNet, which attains the highest scores in Dice(94.34%), and IoU(89.28%). However, PRaNet[28] scores the highest in Precision(96.57%), while DUCK-Net[21] scores highest in Recall(93.92%) and Accuracy(99.29%). These results underscore the efficacy of the proposed MingleNet architecture in semantic segmentation tasks, outperforming other segmentation models in Dice and IoU.

Table 7. Segmentation Accuracy of our model on CVC-ColonDB compared to other models.

CVC-ColonDB					
Model	Dice	IoU	Precision	Recall	Accuracy
U-Net [4]	86.50%	76.2%	94.34%	79.94%	97.99%
U-Net [4] with dynamic data augmentation	82.12%	69.67%	92.12%	74.09%	97.39%
HRNetV2 (pre-trained)[24, 25]	63.83%	46.87%	58.58%	70.10%	95.65%
HarDNet-DFUS (pre-trained)[30]	73.98%	58.70%	95.00%	60.57%	97.61%
MSRF-Net (pre-trained)[23]	83.71%	71.98%	86.03%	81.51%	98.29%
Meta-Polyp (pre-trained) [31]	86.70%	79.00%	-	-	-
FCN-Transformer (pre-trained)[29]	90.73%	83.04%	91.07%	90.40%	98.99%
PraNet(pre-trained)[28]	91.31%	84.01%	**96.57%**	86.59%	99.01%
DUCK-Net (pre-trained)[21]	93.53%	87.85%	93.14%	**93.92%**	**99.29%**
MingleNet	**94.34%**	**89.28%**	95.89%	92.84%	99.10%

The best results are in **bold**.

Discussion

This study shows the performance of ensemble learning over individual base models. Ensemble learning uses the collaborative strength of these base models, where each model adds its unique insights. This approach improves accuracy and robustness significantly. Our ensemble learning model MingleNet outperformed each base model consistently. The individual base models did well in some aspects of the task, but they had limitations in handling the full complexity of the benchmarks. Ensemble learning fills these gaps by using the diverse patterns captured by each model, leading to a better understanding of the data.

Additionally, we show that dynamic data augmentation improves the base models' performance. Dynamic data augmentation adds augmentations to the training input before each training epoch. These data variations create diverse datasets for training models. The diversity in training data enhances the model performance.

MingleNet performs well on medical image segmentation. According to paperswithcode [17], MingleNet ranks **8th** in Dice (**93.19%**) and **17th** in IoU (**87.24%**) on the Kvasir-SEG benchmark. On the CVC-ClinicDB[15] and CVC-ColonDB[16] benchmarks, MingleNet ranks **1st** in Dice (**95.99% and 94.34%**) and IoU (**92.29% and 89.28%**) respectively.

This study shows that using weaker models at the second layer (U-Net [4] and DeepLab[6], weaker than DoubleU-Net[11] and DeepLabv3+[10]) improves the first layer's result slightly. This is because the second layer models use the average of the first layer as the fourth channel. The fourth channel acts as a hint for the second layer models. Moreover, averaging deep learning models improves their accuracy. This is because each model has its own strengths and weaknesses, and averaging them can overcome their weaknesses.

References

[1] R. Wang, T. Lei, R. Cui, B. Zhang, H. Meng, and A. K. Nandi, “Medical image segmentation using deep learning: A survey” IET Image Processing, vol 16, no. 5, pp. 1243-1267, 2022.
doi: https://doi.org/10.1049/ipr2.12419

[2] R. M. Haralick and L. G. Shapiro, “Image segmentation techniques” Computer Vision, Graphics, and Image Processing, vol. 29, no. 1, pp. 100–132, 1985.
doi: https://doi.org/10.1016/S0734-189X(85)90153-7

[3] D. L. Pham, C. Xu, and J. L. Prince, “Current Methods in Medical Image Segmentation” Annual Review of Biomedical Engineering, vol. 2, no. 1, pp. 315–337, 2000.
doi: https://doi.org/10.1146/annurev.bioeng.2.1.315

[4] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”Lecture Notes in Computer Science, vol. 9351, pp. 234–241, 2015.
doi: https://doi.org/10.1007/978-3-319-24574-4_28

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 12, pp. 2481-2495, 2017.
doi: 10.1109/TPAMI.2016.2644615

[6] 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 Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834–848, 2018.
doi: https://doi.org/10.1109/tpami.2017.2699184

[7] J. Chen, Y. Lu, and C. Xu, "TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation", 2021.
doi: https://doi.org/10.48550/arXiv.2102.04306

[8] Z. Zhou, M.R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A Nested U-Net Architecture for Medical Image Segmentation",Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, pp. 3-11, 2018.
doi: https://doi.org/10.1007/978-3-030-00889-5_1

[9] F. Milletari, N. Navab, and S.A. Ahmadi, "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation" 2016 Fourth International Conference on 3D vision (3DV), pp. 565-571, 2016.
doi:https://doi.ieeecomputersociety.org/10.1109/3DV.2016.79

[10]LC. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam," Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation"Proceedings of the European Conference on Computer Vision (ECCV)vol 11211,pp.833–851, 2018.
doi:https://doi.org/10.1007/978-3-030-01234-2_49

[11] D. Jha, M. Riegler, D. Johansen, P. Halvorsen, and H. Johansen, "DoubleU-Net: A Deep Convolutional Neural Network for Medical Image Segmentation" 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS) pp. 558-564, 2020.
doi: https://doi.org/10.48550/arXiv.2006.04868

[12] D. Opitz and R. Maclin, "Popular Ensemble Methods: An Empirical Study" Journal of Artificial Intelligence Research, vol. 11, no. 1, pp. 169–198, 1999.
doi: https://doi.org/10.1613/jair.614

[13] P. Proscura and A. Zaytsev, "Effective training-time stacking for ensembling of deep neural networks" Proceedings of the 2022 5th International Conference on

Artificial Intelligence and Pattern Recognition, pp. 78-82, 2023.
doi: https://doi.org/10.1145/3573942.3573954

[14] 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." MultiMedia Modeling: 26th International Conference, MMM 2020, Daejeon, South Korea, January 5–8, 2020, Proceedings, Part II 26.pp. 451-462, 2020.
doi: https://doi.org/10.1007/978-3-030-37734-2_37

[15] J.Bernal, F. J. Sánchez , G. Fernández-Esparrach, D. Gil, C. Rodríguez, and F. Vilariño, "WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians." Computerized Medical Imaging and Graphics, vol. 43, pp. 99-111, 2015.
doi: https://doi.org/10.1016/j.compmedimag.2015.02.007

[16] D. Vazquez, J.Bernal, F.J. Sánchez ,G. Fernández-Esparrach, A.M. Lopez, A. Romero, M. Drozdzal, and A. Courville, "A benchmark for endoluminal scene segmentation of colonoscopy images"Journal of healthcare engineering, vol. 2017, 2017.
doi:https://doi.org/10.1155/2017/4037190

[17] Papers with Code. Benchmark (Medical Image Segmentation). Retrieved September 14, 2023
Available: https://paperswithcode.com/task/medical-image-segmentation

[18] V. Thambawita, S. A. Hicks, P. Halvorsen, and M. A. Riegler, "DivergentNets: Medical Image Segmentation by Network Ensemble," EndoCV@ISBI, 2021.
Available:https://api.semanticscholar.org/CorpusID:235614237

[19] E. Sanderson and B.J. Matuszewski , "FCN-Transformer Feature Fusion for Polyp Segmentation." Annual Conference on Medical Image Understanding and Analysis, pp. 892-907, 2022.
doi: https://doi.org/10.1007/978-3-031-12053-4_65

[20] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M.Druzhinin, and A. A. Kalinin, “Albumentations: Fast and Flexible Image Augmentations” Information, vol.11, no. 2, pp. 125, 2020.
doi: https://doi.org/10.3390/info11020125

[21] R.G. Dumitru, D. Peteleaza, and C. Craciun, “Using DUCK-Net for polyp image segmentation” Scientific Reports, vol. 13, no. 1, 2023.
doi: https://doi.org/10.1038/s41598-023-36940-5

[22] J. Lever, M.Krzywinski, and N.Altman, “Classification evaluation,” Nature Methods, vol. 13, no. 8, pp. 603–604, 2016.
doi: https://doi.org/10.1038/nmeth.3945

[23] A. Srivastava, “MSRF-Net: A Multi-Scale Residual Fusion Network for Biomedical Image Segmentation” IEEE Journal of Biomedical and Health Informatics, vol. 26, no. 5, pp. 2252-2263, 2022.
doi: https://doi.org/10.1109/JBHI.2021.3138024

[24] K. Sun , B. Xiao, D. Liu, and J. Wang, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.5693-5703, 2019.

[25] K. Sun, Y. Zhao, B. Jiang,T. Cheng, B. Xiao, D.Liu, Y. Mu, X. Wang, W.Y. Liu, and J.D. Wang, “High-resolution representations for labeling pixels and regions.”, 2019.
doi: https://doi.org/10.48550/arXiv.1904.04514

[26] 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” published 2021 in Pattern Recognition. ICPR International Workshops and Challenges,vol.12668,pp 307–314, 2021.
doi: https://doi.org/10.48550/arXiv.2012.15245

[27] N.K. Tomar, A. Shergill, B. Rieders, U. Bagci, and D. Jha, “TransResU-Net: Transformer based ResU-Net for real-time colonoscopy polyp segmentation”, 2022.
doi:https://doi.org/10.48550/arXiv.2206.08985

[28] D.P Fan, G.P. Ji , T. Zhou ,G. Chen, H.Z. Fu , J.B. Shen and S.Ling, “PraNet: Parallel Reverse Attention Network for Polyp Segmentation.” Medical Image Computing and Computer Assisted Intervention. vol.12266, pp.263-273, 2020.
doi: https://doi.org/10.1007/978-3-030-59725-2_26

[29] E. Sanderson and B.J. Matuszewski, “FCN-Transformer Feature Fusion for Polyp Segmentation.”CB. (eds) Medical Image Understanding and Analysis,vol 13413,pp.892-907, 2022.
doi: https://doi.org/10.1007/978-3-031-12053-4_65

[30] T. Y. Liao, C. H. Yang, Y. W. Lo, K. Y. Lai, P. H. Shen, and Y. L. Lin “HarDNet-DFUS: An Enhanced Harmonically-Connected Network for Diabetic Foot Ulcer Image Segmentation and Colonoscopy Polyp Segmentation”, 2022.
doi: https://arxiv.org/abs/2209.07313

[31] Q.H. Trinh, “Meta-Polyp: a baseline for efficient Polyp segmentation” in 2023 IEEE 36th International Symposium on Computer-Based Medical Systems, pp.742-747, 2023.
doi: https://doi.org/10.48550/arXiv.2305.07848