Dual Metric Learning for Few-Shot Weakly-Supervised Optic Disc and Cup Segmentation on Fundus Images

Pandega Abyan Zumarsyah [1], Igi Ardiyanto [1], Kazuhiko Hamamoto [2], and Hanung Adi Nugroho [1,*]

[1] Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada, Yogyakarta, Indonesia
[2] Department of Information Media Technology, School of Information Science and Technology, Tokai University, Kanagawa, Japan
Email: pandegaabyanzumarsyah@mail.ugm.ac.id (P.A.Z.); igi@ugm.ac.id (I.A.); hamamoto@tokai.ac.jp (K.H.); adinugroho@ugm.ac.id (H.A.N.)
*Corresponding author

***Abstract*—Segmentation of the Optic Disc (OD) and Optic Cup (OC) on fundus images is vital for glaucoma diagnosis. Deep learning has been utilized to automate this segmentation task. However, it typically requires a large number of labeled images. To address this issue, we propose Dual Metric Learning for OD and OC Segmentation (DMLOS), which requires only a few images with some of their pixels labeled. It consists of a neural network for embedding extraction, followed by dual branches to obtain prototypes and predictions. The Omni Training algorithm is used to improve data utilization and use a diverse number of shots. Meanwhile, DeepLabv3+ and miniUNet are the neural network used. We extensively evaluated DMLOS on the DRISHTI-GS, REFUGE, and RIM-ONE r3 datasets using various numbers of shots and label densities. Using 15 shots with 0.1 grid density, DMLOS achieved an Intersection over Union of 92.56% for OD and 73.08% for OC on DRISHTI-GS, surpassing other less-supervised methods. The results demonstrate the potential of DMLOS as an effective approach in low-label scenarios.**

***Keywords*—few-shot, weakly-supervised, prototypes, sparse label, fundus image, segmentation, glaucoma**

I. Introduction

Glaucoma is the second leading cause of blindness worldwide [1]. Early diagnosis is essential to prevent the damage from worsening [2, 3]. Diagnosis usually involves segmenting parts of the retinal fundus images, namely the Optic Disc (OD) and Optic Cup (OC) [2, 4, 5]. However, it is often complicated, slow, and expensive when done manually [2–4, 6].

Several studies have used deep learning for automatic OD and OC segmentation [3, 6, 7]. While they achieve high accuracy, common deep learning methods require a large number of images with dense labels, where each pixel in the image is labeled [2, 3, 6, 7]. For OD and OC segmentation, a fundus image requires dense labeling by experts for approximately 2 min [8], and deep learning requires hundreds of such images. Thus, preparing numerous densely labeled fundus images is costly and time-consuming [2, 6].

While there are large public datasets, common deep learning methods perform poorly on new datasets with a different distribution from their training data [9–11]. Variability in the distribution of fundus image datasets is unavoidable due to the use of diverse imaging equipment [12]. Therefore, utilizing large public datasets to train common deep learning models is insufficient.

To address the issue of expensive fully labeled data, there are various approaches to adapt deep learning for limited labeled data. In many of these approaches, there are target data, the limited labeled data that are dealt with, and source data, the more easily obtained labeled data [13]. The ideal approach in the medical field can utilize target data with limited labels while still achieving satisfactory performance. In addition, it should be lightweight and resource-efficient [14] to be used in low-resource medical facilities [15, 16].

Transfer Learning (TL) is the simplest approach [17], but it is ineffective when the target dataset is too limited or too different from the source dataset [18, 19]. Few-Shot Segmentation (FSS) is an alternative to TL that can utilize a few labeled target images, but the labels should be dense. Meanwhile, Weakly-Supervised Segmentation (WSS) can utilize weak labels, but it requires a large number of labeled images. Another approach is Unsupervised Domain Adaptation (UDA), but it requires full retraining each time the target datasets are changed [10].

In addition to these approaches, a promising approach called Few-shot Weakly-supervised Segmentation (FWS) has emerged. As the name suggests, FWS combines FSS and WSS. Therefore, FWS can utilize a few sparsely labeled target images [20]. Both UDA and FWS usually train a model on public datasets, but after adaptation using

Manuscript received October 19, 2025; revised October 27, 2025; accepted December 29, 2025; published March 10, 2026.

limited labeled images, the model can perform well on new target datasets with a different distribution [10, 20]. Compared with UDA, FWS does not require full retraining when the target datasets are changed [20].

Despite its potential, only a few studies have performed few-shot weakly-supervised OD and OC segmentation on fundus images [21, 22].

The main objective of this study is to develop a metric-based FWS method for few-shot weakly-supervised OD and OC segmentation on fundus images. We refer to it as Dual Metric Learning for OD and OC Segmentation (DMLOS). The metric approach involves generating class prototypes from the support data and then comparing them with the query images via metric functions [23]. The metric approach was chosen because it is the most commonly used approach for FSS [23] and is used in some FWS studies [24–27]. It is also lighter and more efficient for low domain shift cases [20]. The contributions of this study include the following:

- Development of a novel dual metric learning method for FWS;
- One of the first studies to perform FWS on retinal fundus images;
- Comprehensive evaluations across multiple datasets, variants, numbers of shots, and sparse label types.

II. Literature Review

A. OD and OC Segmentation for Glaucoma Diagnosis

There are multiple techniques for glaucoma diagnosis [1, 3]. Fundus imaging is an easy, non-invasive, and useful technique [3]. Various parts of the retina can be analyzed from a fundus image [3]. Fig. 1 presents an example of a fundus image from the DRISHTI-GS dataset [28].

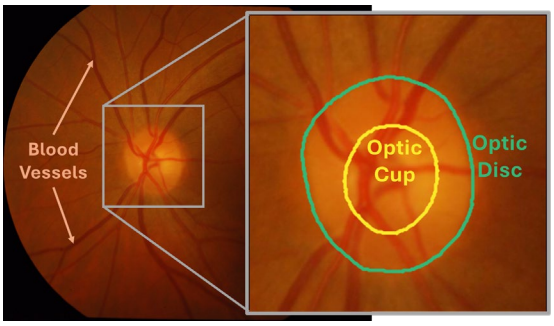

Fig. 1. Sample of fundus image with highlighted Optic Cup (yellow) and Optic Disc (green).

The two essential parts for diagnosing glaucoma are OD and OC. As presented in Fig. 1, OD is the yellowish part of the retina where blood vessels and neurons converge, while OC is the brighter part in the center of the OD [29].

Various analyses can be performed based on the size and shape of OD and OC. The measurement of the Cup-to-Disc Ratio (CDR), the ratio between OC and OD sizes, has become a widely used analysis [2, 4, 5, 30]. In glaucoma cases, the OC is larger than it should be, sometimes even almost covering the entire OD [29–31]. Thus, glaucoma cases have higher CDR. Measurement of CDR requires OD and OC segmentation [2, 4, 5, 30]. This segmentation is often complicated, slow, and expensive if performed manually by eye specialists [2–4, 6].

B. Deep Learning with Limited Data for OD and OC Segmentation

This section discusses various approaches and studies using deep learning with limited data for OD and OC segmentation. Each approach has its own limitations, and FWS is a promising alternative to address these limitations. However, we learned from various approaches to adopt suitable techniques in our FWS method.

TL trains the model on the source data, then fine-tunes it on the target data [5]. It is simple and effective when the target data have sufficient labeled images. However, studies using TL for OD and OC segmentation required hundreds of labeled target images [5, 30]. TL also loses its effectiveness when the target dataset is very different from the source [18, 19]. Moreover, our previous experiment demonstrated that TL using natural image datasets such as ImageNet [32] is not very helpful for medical image segmentation. Therefore, we did not use TL in this study.

FSS does not require numerous labeled target images, but the labels should be dense. Therefore, the annotator only needs to focus on a few images. Two studies have used FSS for OD and OC segmentation. OSAM-Fundus utilized both DINOv2 and Segment Anything Model (SAM) to perform training-free one-shot segmentation [33]. Another study is a Prototype-based Feature Mapping Network (PFMNet) that involves extracting prototypes from features, transforming the prototypes, and comparing the prototypes with other features to obtain predictions. Our study adopted the idea of a training-free network in OSAM-Fundus and a prototype-based network similar to PFMNet. Compared with these FSS studies, our study used weak labels rather than costly dense labels.

Instead of utilizing a few images with dense labels, WSS utilizes weak labels but with numerous images. Therefore, the annotation effort is medium as in FSS. There are multiple types of weak labels used in WSS, but the most common ones in OD and OC segmentation are bounding boxes [34, 35] and image-level classes [35, 36]. One weak label type that is less common but promising for OD and OC segmentation is sparse labels, where only a few pixels are labeled. These sparse labels can be of various types, such as points [37, 38] and scribbles [37–39]. Compared with other weak labels, sparse labels achieved the closest performance to dense labels [40]. Therefore, we utilized a few images labeled with points, scribbles, and other sparse label types.

Surprisingly, one approach that is common for OD and OC segmentation is UDA. Although it does not require any labels on the target data, it requires labeled source images and full retraining when the target data change [10]. The retraining requires both the source and target datasets [41, 42]. Therefore, UDA is suitable when the computational cost is much lower than the annotation cost. Usually, UDA involves adversarial learning techniques [10, 43–48] with segmentor and discriminator networks. Some studies have also utilized image

synthesis [43, 44], consistency loss [45, 48], and mean teacher networks [45, 47]. Due to the need for retraining and the complex architecture, we did not utilize UDA. However, we used the same dataset utilization as many UDA methods to compare the results.

C. Few-Shot Weakly-Supervised Segmentation Methods

Although only a few FWS studies that perform OD and OC segmentation, multiple studies are performing FWS, ranging from the earliest to the latest: Prototype Alignment Network (PANet) [26], Weakly-supervised Segmentation Learning (WeaSeL) [49], Base and Meta (BAM) [50], Holistic Prototype Activation (HPA) [51], ProtoSeg [24], Pseudo-labeling Texture Segmentation (PTS) [25], and Adversarial Mining Transformer (AMFormer) [27].

PANet [26] is one of the earliest and most popular FSS methods, but it is designed for natural images. PANet introduced Prototype Alignment Regularization (PAR) between the support and query [26]. BAM [50] and HPA [51] outperform PANet on natural images, but they require a model that is pre-trained on large datasets. There is also AMFormer [27], a recent FSS method that does not require pre-trained models but does require a complex multi-stage adversarial training. PTS [25] also utilized multi-stage training and has been evaluated on fundus images, but the task is blood vessel segmentation, and its performance on this task is suboptimal.

These methods are designed for FSS, but one of their evaluations involves FWS. They are designed for natural images and have not been evaluated on medical cases. The only exception is PTS [25] which is designed for microstructure images and achieves suboptimal performance on medical images. In addition, each of them requires either models that are pre-trained on large datasets [26, 50, 51] or multi-stage training [25, 27]. The former is less suitable for medical images due to the lack of large datasets, whereas the latter is challenging and resource-consuming. Therefore, these methods are not suitable for medical cases, despite the good performance in others.

A few studies have developed FWS methods for medical images, such as WeaSeL [49] and ProtoSeg [24]. WeaSeL utilizes double gradient optimization, which is computationally expensive [20]. Meanwhile, ProtoSeg extracted prototypes and utilized them to obtain predictions using a metric function [24]. Both have been evaluated on multiple types of medical images, but are limited to binary segmentation on grayscale images.

Compared with the existing methods, our method is lighter, does not require pre-trained models or multi-stage training, and is able to handle multiclass segmentation on multichannel images. Therefore, our method is more suitable for medical images. The method is also designed for FWS and extensively evaluated on fundus images.

In previous studies, we have performed FWS on OD and OC segmentation by adapting WeaSeL and ProtoSeg for multiclass segmentation on multichannel images [21, 22]. We also evaluated multiple training strategies. In this study, we developed a novel and better FWS method by incorporating insights from previous FWS methods.

Previous studies suggest that in cases with low domain shift, as in this case, metric methods like ProtoSeg are more suitable [20, 22]. However, the prototypical network in ProtoSeg is prone to an inconsistent learned embedding space, which can lead to poor generalization. To address this, we designed the dual metric learning based on the idea of PAR in PANet [26], which is another metric method. Both PANet and ProtoSeg [24, 26] employ a lightweight metric approach suitable for our study. In addition to the dual metric learning, we developed a training algorithm based on the results of our preliminary study [21].

III. Materials and Methods

A. Datasets

We used three datasets: DRISHTI-GS [28], REFUGE [52], and RIM-ONE r3 [53]. We selected them because previous studies with limited labeled data also used them [10, 34, 43–48].

Table I presents the information on the datasets used in this study. Each dataset has a different camera, field of view, resolution, and distribution of glaucoma and normal classes. Note that we treated the REFUGE train and the REFUGE validation + test as different datasets. To avoid data leakage, we performed splitting before other processing. REFUGE train is used for training, while REFUGE validation, DRISHTI-GS train, and RIM-ONE r3 train are used for validation. The final test used REFUGE test, DRISHTI-GS test, and RIM-ONE r3 test.

For each dataset, cropping was applied to the fundus images and their corresponding labels to focus on the OD area. In addition, all images are resized to 256 × 256 for easier processing by the model.

TABLE I. Acquisition Details and Class Distribution of the Used Datasets

Dataset	Acquisition			Class	
	Camera	FOV	Resolution	G	N
REF train	Zeiss Visucam 500	-	2124×2056	40	360
REF val + test	Canon CR-2	-	1634×1634	80	720
DGS	-	30°	2896×1944	70	31
RO3	Kowa WX 3D	34°	2144×1424	74	85

Note: FOV: Field of View, G: Glaucoma, N: Normal, REF: REFUGE, val: validation, DGS: DRISHTI-GS, RO3: RIM-ONE r3

All datasets have numerous images with dense OD and OC segmentation labels. However, we transformed them to simulate datasets with a few sparsely labeled images. The transformation involves simulation of various sparse labels and batch preparation with diverse numbers of shots. The variability makes the model more robust and allows us to analyze the recommended configuration for tuning. This transformation is done during the training and is discussed further in the next sections.

B. Sparse Label Simulation

We simulated various types of sparse labels from their corresponding dense labels to perform comprehensive evaluations across multiple types of sparse labels. With the comprehensive evaluations, the most recommended sparse label type can be found.

There are five sparse label types in this study, similar to those by Gama *et al.* [24], but with improvements. The main improvement is the adaptation for multiclass labels. In addition, since annotators are unlikely to annotate just a single pixel per point, we improved the points and grid so that their point size can be larger and more realistic. Each sparse label has a density parameter representing the number of labeled pixels. Fig. 2 presents the simulated sparse labels with various density values.

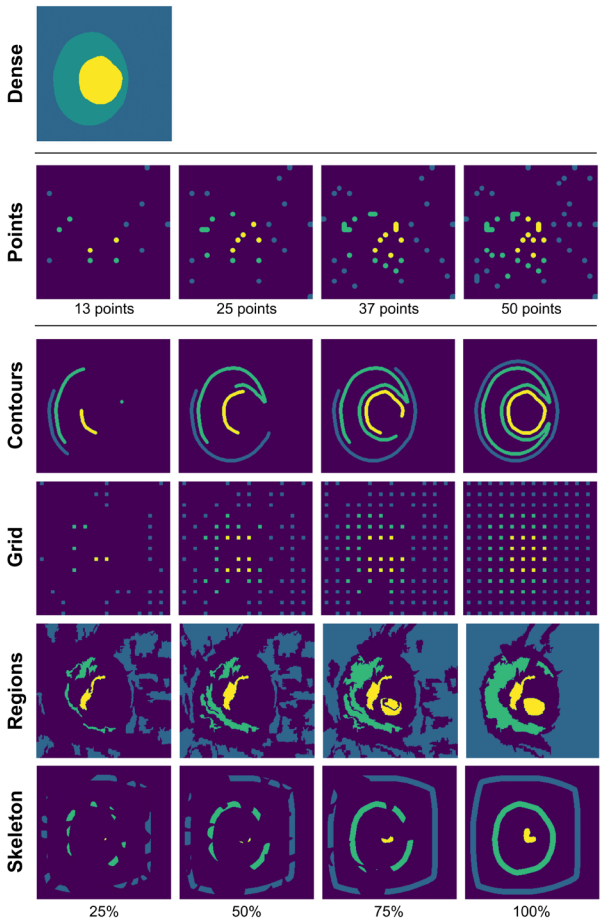

Fig. 2. Simulated sparse labels and their original dense label. The four classes: background (teal blue), Optic Disc (green), Optic Cup (yellow), and unannotated (purple).

1) *Points*

Points are a natural way for humans to indicate objects [38] and have been commonly used as weak labels. Because of its simplicity and popularity, we utilized points in this study. The label is simulated by selecting n_p random pixels from the dense label. The number of points n_p is the density parameter of this label.

To allow a larger point size, the original dense label is downscaled, and then the obtained sparse label is upscaled. Then, morphological dilation is applied for enhancement.

2) *Contours*

This label simulates drawing scribbles on the inner and outer parts of objects. Scribbles are a user-friendly alternative to other weak labels [40]. They are useful for cases with a single connected object [24], as in this study.

We simulated it by eroding the objects, followed by finding the contours of the eroded objects with a marching squares algorithm. Subsequently, some p_c percentage of the contours is randomly selected as the sparse label. This resembles annotators who only draw some lines, rather than the entire contour. Therefore, the p_c is the density parameter of this sparse label. Finally, to allow a larger line width, the sparse label is dilated.

3) *Grid*

The grid label represents the annotation of some points provided in a grid pattern. We utilized this label because it has been proven to be highly reliable in another study [24].

Initially, a grid is created, where each point is separated by a certain number of pixels. All points in the grid are included as the initial sparse label.

To simulate the variable density, rounded binary blobs are generated to cover p_g percentage of the sparse label. The final sparse label includes only the points covered by the blobs. This resembles annotators who only mark some parts of the image. Therefore, p_g is the density parameter of this sparse label.

For a larger point size, this label takes the same approach as the points label. Downscaling is performed before grid creation, while upscaling is conducted after the initial sparse label is generated.

4) *Regions*

This label simulates annotators assigning a class to a few pre-generated superpixels or regions. It is utilized because it can cover many pixels and achieve accurate predictions with simple and quick annotations [24].

The regions are generated using the Simple Linear Iterative Clustering (SLIC) algorithm [54]. SLIC is chosen because it has computational and performance advantages over other algorithms [55]. After the generation, regions that cover multiple classes are discarded. To simulate the variable density, as in the grid label, blobs covering p_r percentage of the label are utilized.

5) *Skeleton*

The skeleton represents drawing scribbles on the center parts of objects. We simulated both the skeleton and contours to assess the effectiveness of scribbles on both the edges and the center parts of objects, respectively.

The skeleton is simulated by applying a skeletonization algorithm to the dense label. Then, the sparse label is dilated to increase the line width, as in the contours label. Blobs are also utilized to simulate the variable density by covering p_s percentage of the label.

C. *FWS Formulation*

We formulated the source and target datasets in FWS based on the definition of FWS by Gama *et al.* [24, 49].

A source dataset $\mathcal{S}$ comprises pairs (x, y) where $x \in \mathbb{R}^{H \times W \times L}$ is an image with dimensions $H \times W$ and L channels. Meanwhile, $y \in \mathbb{R}^{H \times W}$ is a dense label whose pixel has a value $c \in \{1,2,\dots,C\}$, which is the class label. The source dataset $\mathcal{S}$ is split into support $\mathcal{S}^{sup}$ and query $\mathcal{S}^{qry}$.

The target dataset $\mathcal{T}$ is also split into support $\mathcal{T}^{sup}$ and query $\mathcal{T}^{qry}$. The support $\mathcal{T}^{sup}$ is also composed of pairs

$(x^{sup}, \dot{y}^{sup})$, but $\dot{y} \in \mathbb{R}^{H \times W}$ is a sparse label instead of a dense label. In addition, the query $\mathcal{T}^{qry}$ is only composed of unlabeled images, where each image is represented as $x^{qry} \in \mathbb{R}^{H \times W \times L}$.

During training, the algorithm uses the source dataset $\mathcal{S}$ to optimize the parameters θ of a neural network Φ. Meanwhile, during inference, it uses x^{qry} and $\mathcal{T}^{sup}$ of the target dataset to obtain the prediction $\hat{y}^{qry}$.

D. Omni Training Algorithm

We trained the DMLOS using our algorithm called Omni Training, which improves data utilization and uses a diverse number of shots during training. This involves transforming the source dataset $\mathcal{S}$ into a new dataset $\mathcal{F}$. As mentioned, the transformation involves batch preparation. The support batches are sampled with diverse sample sizes while the query ones are sampled with a fixed sample size. The sampling is performed randomly without replacement to ensure that all data are used equally. We also ensured that no leakage exists between the support and query batches by doing the splitting before the batch preparation. Thus, with Omni Training, the data are optimally used, and the model can learn from various numbers of shots.

Algorithm 1 presents the Omni Training algorithm that runs for n_{ep} epochs. Initially, the parameters are initialized and the dataset $\mathcal{S}$ is transformed into a new dataset $\mathcal{F}$. A configuration F determines how the transformation is performed, including the possible number of shots and sparse label density values. Each epoch then iterates over the dataset $\mathcal{F}$ where each iteration takes four batches: support images X^{sup}, support sparse labels $\dot{Y}^{sup}$, query images X^{qry}, and query dense labels Y^{qry}. The dual metric learning algorithm then processes the four batches and returns the loss $\mathcal{L}$. The gradient of the loss $\mathcal{L}$ is then used to update the parameters via an optimizer algorithm opt().

Algorithm 1. Omni Training

Require: $\mathcal{S}$, n_{ep}, F
Initialize θ
Transform $\mathcal{S}$ into $\mathcal{F}$ based on F
For $e = 1$ to n_{ep} do
  For $X^{sup}, \dot{Y}^{sup}, X^{qry}, Y^{qry}$ in $\mathcal{F}$ do
    Run Dual Metric Learning on $X^{sup}, \dot{Y}^{sup}, X^{qry}, Y^{qry}$ to get $\mathcal{L}$
    Update $\theta \leftarrow \theta - \text{opt}(\nabla_\theta \mathcal{L})$
  End For
End For

E. Dual Metric Learning Algorithm

This algorithm is the core of the DMLOS. It dictates how the model Φ is trained using the images and labels from the support and query sets. In addition to the four batches, the algorithm takes a regularization parameter λ to balance the contribution of the losses. The algorithm starts by extracting the embeddings of support E^{sup} and query E^{qry} with depth M via a neural network Φ:

$$E = \Phi_\theta^M(X) \tag{1}$$

Then, the support embeddings E^{sup} and the support sparse labels $\dot{Y}^{sup}$ are used to calculate the class prototypes $\mathcal{P}_c$ for each class c:

$$\mathcal{P}_c = \frac{\sum_k^B \sum_j^N \dot{Y}_c^{jk} \odot E^{jk}}{\sum_k^B N_c^k} \tag{2}$$

In Eq. (2), B is the batch size, $N = H \times N$ is the number of pixels in the image, and $\odot$ is the element-wise multiplication. Meanwhile, N_c is an array with B elements, where each element stores the number of annotated pixels for a class c in one image from the batch.

After calculating the class prototypes, the query embeddings are compared with them using the Euclidean distance $d(\cdot)$. Softmax is then applied to obtain the class probabilities ρ_c:

$$\rho_c^j = \frac{exp(-d(E^j, \mathcal{P}_c))}{\sum_i^C exp(-d(E^j, \mathcal{P}_i))} \tag{3}$$

Next, the Support-to-Query (S2Q) loss $\mathcal{L}_{S2Q}$ is calculated using the class probabilities ρ_c and the query labels Y^{qry}:

$$\mathcal{L} = \frac{1}{N} \sum_j^M \sum_i^C Y_i^j log(\rho_i^j) \tag{4}$$

The query predictions $\hat{Y}^j$ can be obtained by selecting the class with the highest probability:

$$\hat{Y}^j = \arg\max_c \rho_c^j \tag{5}$$

After all of that, the processes for obtaining the class prototypes, class probabilities, and segmentation loss are then performed once again. However, the roles of the support and query sets are switched. Instead of support sparse labels, the query predictions are used to calculate the class prototypes. Similarly, instead of query dense labels, the support sparse labels are used to calculate the Query-to-Support (Q2S) loss $\mathcal{L}_{Q2S}$. Finally, the two losses are added with λ controlling the contribution of $\mathcal{L}_{Q2S}$.

Algorithm 2 summarizes the complete steps of the proposed DMLOS. Additionally, we present Fig. 3 to illustrate the algorithm. Both support-to-query and query-to-support branches obtain class prototypes, predictions, and segmentation losses. Both branches work together so that the model learns embeddings that align the support and query prototypes, similar to the PAR in PANet [26].

Algorithm 2. Dual Metric Learning

Require: $X^{sup}, \dot{Y}^{sup}, X^{qry}, Y^{qry}, \lambda$
Embeddings Extraction
  Extract E^{sup} from X^{sup} using (1)
  Extract E^{qry} from X^{qry} using (1)
Support-to-Query Branch
  Calculate $\mathcal{P}_c^{sup}$ for each c from $(E^{sup}, \dot{Y}^{sup})$ using (2)
  Calculate ρ_c^{qry} for each c from $(E^{qry}, \mathcal{P}_c^{sup})$ using (3)
  Calculate $\mathcal{L}_{S2Q}$ from (Y^{qry}, ρ_c^{qry}) using (4)
  Predict $\hat{Y}^{qry}$ from ρ_c^{qry} using (5)
Query-to-Support Branch
  Calculate $\mathcal{P}_c^{qry}$ for each c from $(E^{qry}, \dot{Y}^{qry})$ using (2)
  Calculate ρ_c^{sup} for each c from $\left(E^{sup}, \mathcal{P}_c^{qry}\right)$ using (3)
  Calculate $\mathcal{L}_{Q2S}$ from $(\dot{Y}^{sup}, \rho_c^{sup})$ using (4)
Combine Losses
  Calculate $\mathcal{L}_T = \mathcal{L}_{S2Q} + \lambda \mathcal{L}_{Q2S}$
  Return $\mathcal{L}_T$

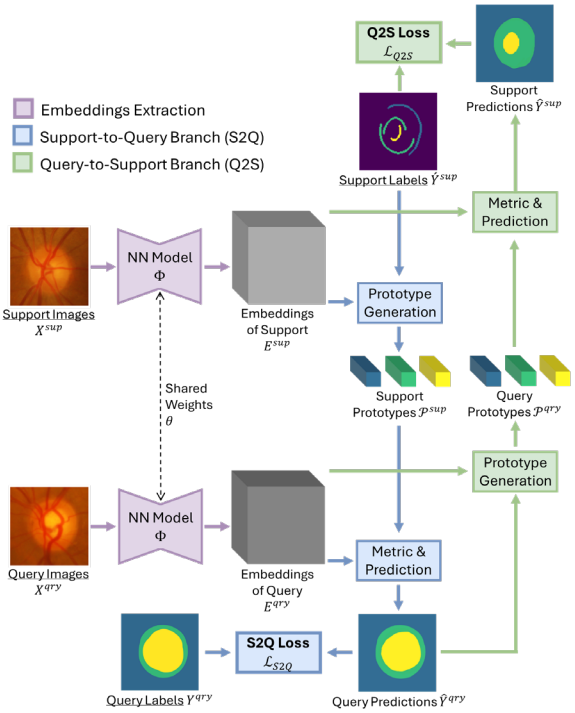

Fig. 3. Illustration of the proposed DMLOS, support & query images with labels (underlined) serve as input while two loss calculations (bold) are the final process.

Regarding the neural network Φ, it can be any image segmentation model. However, instead of returning the final segmentation predictions, the model returns the embeddings. In this study, we evaluated two types of models. The first one is a miniUNet as used by Gama *et al.* in WeaSeL and ProtoSeg [24, 49]. The second model is DeepLabv3+ [56], one of the best-performing models for semantic segmentation. For DeepLabv3+, we evaluated multiple backbones to determine the best one: ResNet50 [57], MobileNetV2 [58], and HRNetv2 [59]. They represent popular, lightweight, and high-resolution backbones, respectively.

In addition, we evaluated the DMLOS with and without the Q2S branch. This is crucial in determining whether the Q2S branch is beneficial.

F. Training and Evaluation Procedures

Similar to standard deep learning, our experiments involve train, validation, and test splits. Each split is divided into support and query. Table II presents the detailed parameters of the support in each split. The parameters are part of the configuration F for transforming the original dataset $\mathcal{S}$ into a new dataset $\mathcal{F}$.

TABLE II. DETAILED PARAMETERS OF SUPPORT IN THE TRAIN, VALIDATION, AND TEST SPLITS

Parameters	Train	Validation	Test
Dataset	REF train	REF val, DGS train, RO3 train	REF test, DGS test, RO3 test
Mode	mix	combine	full combine
Shots	(1, 20)	[5, 10, 15]	[1, 5, 10, 15, 20]
Point Density	(5, 50)	[13, 25, 37]	[1, 13, 25, 37, 50]
Other Density	(0.1, 1.0)	[0.25, 0.5, 0.75]	[0.1, 0.25, 0.5, 0.75, 1.0]

Note: REF: REFUGE, val: validation, DGS: DRISHTI-GS, RO3: RIM-ONE r3

In Table II, the parentheses "()" represent a range with low and high limits. Meanwhile, the square brackets "[]" represent value options.

We used DRISHTI-GS, REFUGE, and RIM-ONE r3, with commonly used train, validation, and test splits. The train split utilizes the "mix" mode that randomly selects the number of shots and the density value. Meanwhile, the "combine" mode used in the validation split generates all possible combinations of the number of shots, sparse label types, and density values. The test split utilizes the "full combine" mode, which is similar to the "combine" mode, but all combinations are evaluated for every query image.

All models were trained using the Adam optimizer [60] with a step scheduler and cross-entropy loss. The primary metric is Intersection over Union (IoU) due to its low bias and popularity [61]. In addition to IoU, we also used a boundary-based metric. This is because boundary errors can affect the CDR measurement, and IoU alone is insufficient to capture such errors. In other words, it is theoretically possible to have a good IoU despite high boundary errors.

In OD and OC segmentation, we focus more on contour or shape than on individual points or outliers. Following the Metrics Reloaded guideline, we found that the Mean Average Surface Distance (MASD) is the most suitable boundary-based metric due to its focus on the contour [62]. Other popular metrics, such as the Hausdorff distance and Boundary IoU, do not respect the contour [62].

IoU and MASD are complementary as they utilize two fundamentally different perspectives: area and boundary. We calculated both metrics using the Metrics Reloaded implementation [62]. The metrics were calculated separately for OD and OC, as in other studies [5, 18].

We performed hyperparameter tuning using Tree-structured Parzen Estimator (TPE) [63] and Hyperband pruner [64] with the objective of maximizing the validation IoU. TPE is effective for deep learning tasks with limited resources and high-dimensional search spaces [65] while Hyperband is the best pruner for TPE.

The tuned values include learning rate, betas and weight decay of the optimizer, gamma of the scheduler, embedding depth M, and regularization parameter λ. After tuning, the best model is evaluated on the test set.

We implemented the proposed method and performed the experiments using PyTorch [66] and PyTorch Lightning [67]. NumPy [68] and scikit-image [69] were used to simulate the sparse labels. Our code is available at https://github.com/pandegaabyan/few-shot-weakly-seg

IV. RESULTS AND DISCUSSION

A. Performance of DMLOS Variants

This section discusses the performance of the four DMLOS variants, which are combinations of the miniUNet or DeepLabv3+ models, with or without Q2S. The chosen backbone for DeepLabv3+ is ResNet, as it achieved better tuning scores than other backbones. For each variant, multiple evaluations were conducted on images from the three datasets with varying numbers of shots, sparse label types, and density values.

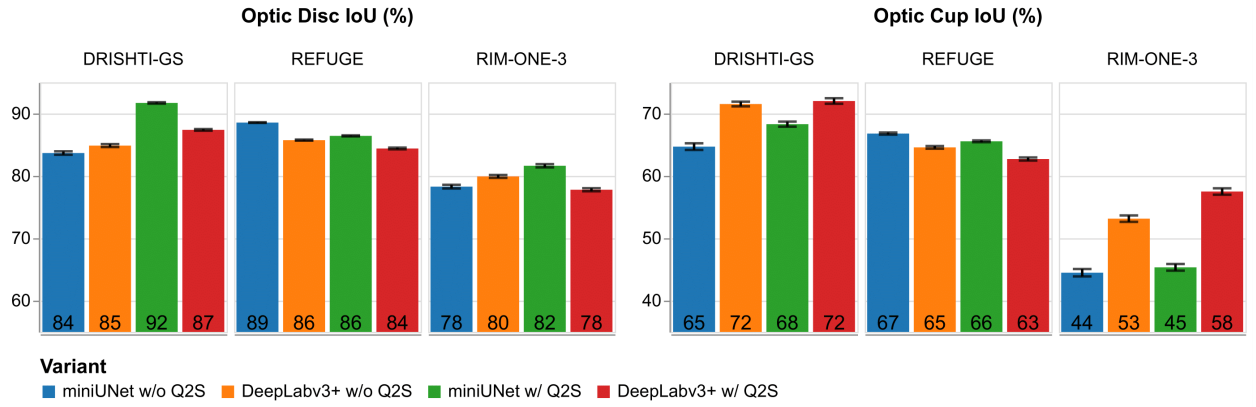

Fig. 4. Intersection over Union (IoU) scores of Optic Disc (left) and Optic Cup (right) segmentation on various variants and datasets averaged across. Higher scores are better.

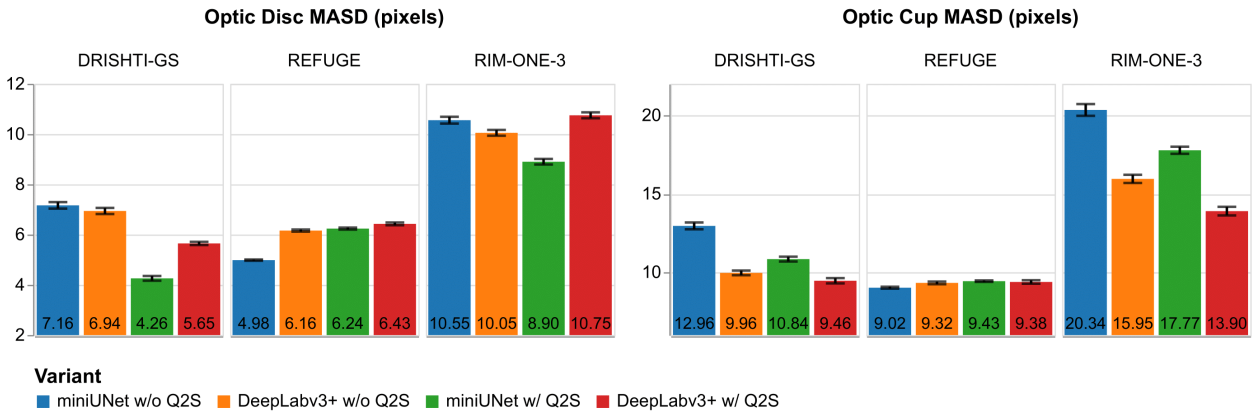

Fig. 5. Mean Average Surface Distance (MASD) scores of Optic Disc (left) and Optic Cup (right) segmentation on various variants and datasets averaged across. Lower scores are better.

Fig. 4 presents the IoU scores of the four variants, while Fig. 5 presents their MASD scores. Generally, IoU and MASD scores are consistent, with higher IoUs correlated with lower MASD scores.

OD consistently achieved higher IoU scores and lower MASD scores than OC. This is because OC is harder to segment than OD. All variants performed worse on RIM-ONE r3 than the other datasets. As will be shown in Section IV.C, other studies have the same finding, suggesting that RIM-ONE r3 is the most difficult dataset.

DeepLabv3+ and Q2S are beneficial for addressing hard segmentation cases. Variants with DeepLabv3+ generally achieved higher OC IoUs and lower OC MASDs, especially on the RIM-ONE r3 dataset. A similar trend also emerges when comparing variants with and without Q2S, although it is less obvious. However, this trend is not observed in the easier OD segmentation. The added complexities likely improved the performance by focusing on hard cases at the cost of a slight decline in easy cases.

Surprisingly, simpler variants often achieved better scores on the REFUGE dataset. DMLOS with miniUNet without Q2S achieved the best scores on the REFUGE dataset, although it obtained the worst scores on other datasets. This is possibly because the REFUGE test dataset is more similar to the train split than other test datasets. The simpler variants focus on the similar dataset instead of becoming generalized.

In addition to the comparison of scores shown in Figs. 4 and 5, we also performed hypothesis tests to compare the mean IoU and mean MASD scores of some pairs of variants. In this case, the mean IoU and mean MASD are averages of OD and OC IoUs and MASDs, respectively.

We used Wilcoxon tests since the data are not normally distributed. Tables III and IV present the *p*-values of the mean IoU and mean MASD tests, respectively.

TABLE III. P-VALUES OF WILCOXON SIGNED-RANK TESTS COMPARING MEAN IOU SCORES OF VARIANTS

Alternative Hypothesis on IoU	Dataset			Conclusion
	DRISHTI-GS	REFUGE	RIM-ONE r3	
DeepLabv3+ w/ Q2S > DeepLabv3+ w/o Q2S	~0.0	1.0	~0.0	→ Q2S improves
miniUNet w/ Q2S > miniUNet w/o Q2S	~0.0	1.0	~0.0	performance
DeepLabv3+ w/ Q2S > miniUNet w/ Q2S	0.9933	1.0	~0.0	→ DeepLabv3+ is
DeepLabv3+ w/o Q2S > miniUNet w/o Q2S	~0.0	1.0	~0.0	better than miniUNet

Note: IoU: Intersection over Union, Q2S: Query-to-Support branch

TABLE IV. P-VALUES OF WILCOXON RANK-SUM TESTS COMPARING MEAN MASD SCORES OF VARIANTS

Alternative Hypothesis on MASD	Dataset			Conclusion
	DRISHTI-GS	REFUGE	RIM-ONE r3	
DeepLabv3+ w/ Q2S < DeepLabv3+ w/o Q2S	~0.0	~0.0	~0.0	→ Q2S improves
miniUNet w/ Q2S < miniUNet w/o Q2S	~0.0	1.0	~0.0	performance
DeepLabv3+ w/ Q2S < miniUNet w/ Q2S	0.57481	~0.0	~0.0	→ DeepLabv3+ is
DeepLabv3+ w/o Q2S < miniUNet w/o Q2S	~0.0	1.0	~0.0	better than miniUNet

Note: MASD: Mean Average Surface Distance, Q2S: Query-to-Support branch

Note that variants with Q2S consistently achieved significantly better performance, except in the REFUGE dataset. In the hard segmentation cases, variants with DeepLabv3+ performed significantly better than those with miniUNet. Otherwise, using the lightweight miniUNet is preferred. These findings are consistent with our previous discussions.

B. *Effects of Shots and Sparse Labels*

Fig. 6 presents the mean IoU scores of the best variant on various numbers of shots and sparse labels. The number of points is divided by 50 to match the other ranges of density values. The error area represents the 95% confidence interval. As expected, higher shots generally resulted in better performance.

Density values play a larger role in a single shot, but have little effect on higher shots. This is because the available information is minimal in a single shot. Thus, additional information from higher density can significantly improve the performance of the proposed method. On higher shots, the available information is already sufficient. Thus, additional information does not significantly improve performance.

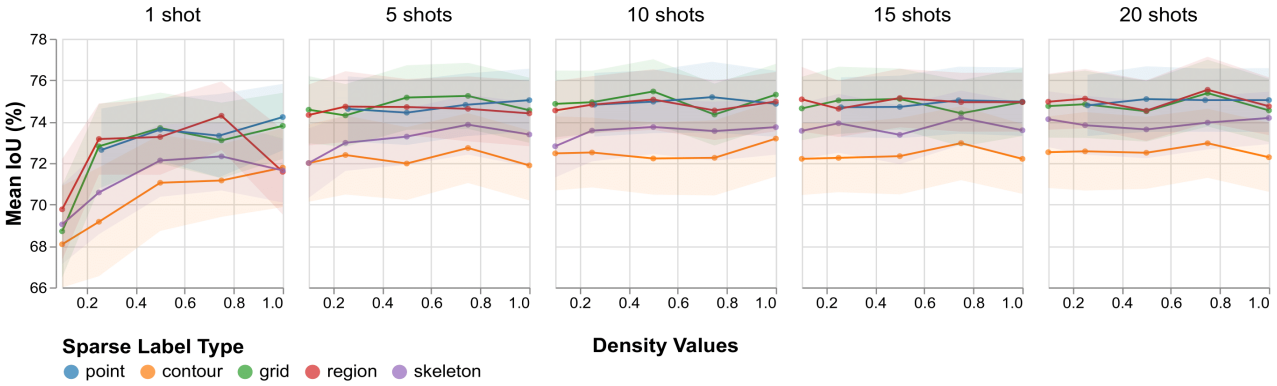

Fig. 6. Mean Intersection over Union (IoU) scores of DeepLabv3+ w/ Q2S variant on various numbers of shots and sparse labels.

Note that the contours label is the worst, followed by the skeleton label, whereas the other three labels achieved a similar good performance. The contours and skeleton labels performed worse, possibly because they captured less diversity. The former only captures the outer parts of the objects, whereas the latter only captures the inner parts. Points and grids can capture more diversity because they can be anywhere, but they require more effort to annotate. Meanwhile, the region label can cover more pixels, and the annotation only requires picking from predefined areas.

Based on this discussion, the region label is the most recommended sparse label due to its good performance and relatively low annotation effort.

Regarding the number of shots, while more shots lead to better performance, using five shots is recommended in cases where annotation is expensive. It is comparable to using larger shots, yet much better than using a single shot.

C. *Comparison with Related Methods*

Table V compares our results with those of related studies. For both miniUNet and DeepLabv3+ variants, we selected one low-shot and one high-shot configuration that achieved high IoU scores.

Most of the studies that utilized the same datasets and performed the same tasks as ours were UDA studies. Few FSS, WSS, or FWS studies perform OD and OC segmentation with the same datasets.

Generally, our DMLOS achieved scores comparable to those of the UDA methods and even outperformed the non-UDA methods.

Although a few UDA methods achieved higher scores, they usually required a heavier backbone. Meanwhile, our DMLOS achieved comparable performance despite using only a miniUNet with fewer than two million parameters. Our DMLOS also does not require any retraining when the target datasets are changed. It only requires extracting prototypes from a few labeled images, which can be done in approximately a tenth of a second.

Compared with WSS and FSS, our DMLOS achieved better performance, although only a limited number of scores were available for comparison. Note that using only five images labeled with a 0.50 grid density, our DMLOS with miniUNet outperformed the WSS methods, which utilized numerous class or bounding box labels, and the FSS methods, which employed 10 densely labeled images.

TABLE V. COMPARISON OF IOU SCORES (%) OF THE BEST DMLOS VARIANTS IN THIS STUDY WITH THOSE OF RELATED STUDIES

Method	Approach	Params	DRISHTI-GS		REFUGE		RIM-ONE r3	
			OD	OC	OD	OC	OD	OC
CFEA [70]	UDA	-	79.78	70.52	88.96	75.86	60.08	46.53
pOSAL [10]	UDA	5.8 M	91.42	72.30	90.83	78.31	76.75	62.59
SIFA [71]	UDA	43.3 M	83.04	57.29	85.69	69.57	74.67	52.84
WGAN [46]	UDA	-	91.20	72.40	-	-	-	-
IOSUDA [44]	UDA	42.8 M	89.53	65.56	91.04	71.03	83.26	60.07
CADA [47]	UDA	9.7 M	80.18	72.41	90.44	77.21	62.13	47.10
SCUDA [72]	UDA	-	90.34	66.61	-	-	84.89	61.65
HyNet (class labels) [36]	WSS	5.3 M	91.09	-	-	-	-	-
GrabCut + UNet (bounding box) [34]	WSS	-	86.37	-	-	-	-	-
OSAM-Fundus (5 shots) [33]	FSS	308 M	91.11	63.41	84.96	61.23	70.42	52.50
OSAM-Fundus (10 shots) [33]	FSS	308 M	91.13	66.31	85.20	62.88	70.97	53.99
MERU (10 shots) [73]	FSS	-	-	-	83.92	61.47	-	-
WeaSeL (20 shots, 0.50 grid) [49]	FWS	1.9 M	65.95	17.35	71.40	29.33	61.08	21.00
ProtoSeg (15 shots, 50 points) [24]	FWS	1.9 M	87.44	68.01	81.29	55.58	82.70	49.71
DMLOS-U (5 shots, 0.50 grid)	FWS	1.9 M	92.82	71.27	86.51	65.64	81.92	46.87
DMLOS-DL3 (5 shots, 0.75 grid)	FWS	39.6 M	88.57	76.08	84.89	64.05	78.27	59.60
DMLOS-U (15 shots, 0.10 grid)	FWS	1.9 M	92.56	73.08	86.46	65.50	81,94	46.52
DMLOS-DL3 (20 shots, 0.75 region)	FWS	39.6 M	88.61	75.10	84.99	65.85	78.04	60.64

Note: IoU: Intersection over Union, Params: number of parameters of the backbone, OD: Optic Disc, OC: Optic Cup, UDA: Unsupervised Domain Adaptation, WSS: Weakly-Supervised Segmentation, FSS: Few-Shot Segmentation, FWS: Few-shot Weakly-supervised Segmentation, -U: miniUNet, -DL3: DeepLabv3+

Although our preliminary study [21] performed FWS on fundus images, the source and target datasets are different. Thus, we cannot compare its results with those of this study. As an alternative, we have implemented WeaSeL and ProtoSeg, then evaluated them with the same dataset utilization as ours. Their configurations, presented in Table V, are those with the highest IoU scores. Our DMLOS outperformed both WeaSeL and ProtoSeg in almost all cases, even when using a lower number of shots.

D. Limitations

This study is still in the prototype or validation stage. Although we have validated the superiority of our method on a crucial medical task with established datasets, the scope of the exploration remains limited. Before applying the method in a real-world setting, more controlled evaluations should be performed on various cases.

There are other medical tasks and datasets with different characteristics. Such tasks may benefit from FWS using our dual metric learning method. Similarly, in addition to DeepLabv3+, other modern models are available. Next studies can explore the method using other modern models on other medical tasks and datasets.

All datasets used in this study are public datasets, which may not represent the challenges of real-world clinical datasets. Moreover, although we strived to generate realistic sparse labels, they may not accurately represent labels from human annotators. Next studies can evaluate the dual metric learning method on other cases with available real-world clinical datasets or sparse labels from human annotators.

V. CONCLUSION

This study presents DMLOS, a dual metric learning for few-shot weakly-supervised OD and OC segmentation. It is trained using the Omni Training algorithm that improves data utilization and uses a diverse number of shots.

The results demonstrate that the dual branch approach outperformed the single branch approach, except in the REFUGE dataset. DeepLabv3+ is useful for hard segmentation cases, but lighter models are preferred for easy cases.

More shots unsurprisingly achieve higher scores, but smaller shots can achieve good scores with effective sparse labels. Points, grid, and region labels are similarly effective, but the region label is more efficient due to the easier annotations. Meanwhile, the contours and skeleton labels are the least effective.

Our DMLOS is comparable to the UDA methods in terms of performance, but it is more efficient because it can be lighter and does not require retraining. Our proposed method is also superior to the FSS and WSS methods, despite using less labeled data.

These results suggest that DMLOS is effective for fundus image segmentation with limited labeled data. With DMLOS, a model can be trained optimally and then adapted to new data using only a few sparsely labeled images.

Next studies could evaluate the dual metric learning method using different model architectures on various datasets and cases.

CONFLICT OF INTEREST

The authors declare no conflict of interest

AUTHOR CONTRIBUTIONS

Pandega Abyan Zumarsyah: Conceptualization, Methodology, Software, Validation, Investigation, Writing—Original Draft, Visualization. Igi Ardiyanto: Conceptualization, Methodology, Writing—Review & Editing, Supervision. Kazuhiko Hamamoto: Conceptualization, Writing—Review & Editing, Supervision. Hanung Adi Nugroho: Conceptualization, Writing—Review & Editing, Supervision, Funding Acquisition. All authors had approved the final version.

FUNDING

This study was funded by the Ministry of Higher Education, Science, and Technology of Indonesia through the Basic Research and the PMDSU program with grant number 2549/UN1/DITLIT/Dit-Lit/PT.01.03/2025.

REFERENCES

[1] R. Raveendran *et al.*, "Current innovations in intraocular pressure monitoring biosensors for diagnosis and treatment of glaucoma—Novel strategies and future perspectives," *Biosensors*, vol. 13, no. 6, 663, June 2023. doi: 10.3390/bios13060663

[2] H. Xiong, S. Liu, R. V. Sharan, E. Coiera, and S. Berkovsky, "Weak label based Bayesian U-Net for optic disc segmentation in fundus images," *Artificial Intelligence in Medicine*, vol. 126, 102261, Apr. 2022. doi: 10.1016/j.artmed.2022.102261

[3] M. J. M. Zedan, M. A. Zulkifley, A. A. Ibrahim, A. M. Moubark, N. A. M. Kamari, and S. R. Abdani, "Automated glaucoma screening and diagnosis based on retinal fundus images using deep learning approaches: A comprehensive review," *Diagnostics*, vol. 13, no. 13, 2180, June 2023. doi: 10.3390/diagnostics13132180

[4] H. Garg, N. Gupta, R. Agrawal, S. Shivani, and B. Sharma, "A real time cloud-based framework for glaucoma screening using EfficientNet," *Multimed. Tools Appl.*, vol. 81, no. 24, pp. 34737–34758, Oct. 2022. doi: 10.1007/s11042-021-11559-8

[5] X. Zhao, S. Wang, J. Zhao, H. Wei, M. Xiao, and N. Ta, "Application of an attention U-Net incorporating transfer learning for optic disc and cup segmentation," *SIViP*, vol. 15, no. 5, pp. 913–921, July 2021. doi: 10.1007/s11760-020-01815-z

[6] X. Ma, G. Cao, and Y. Chen, "A review of optic disc and optic cup segmentation based on fundus images," *IET Image Processing*, vol. 18, no. 10, pp. 2521–2539, 2024. doi: 10.1049/ipr2.13115

[7] M. Alawad *et al.*, "Machine learning and deep learning techniques for optic disc and cup segmentation—A review," *OPTH*, vol. 16, pp. 747–764, Mar. 2022. doi: 10.2147/OPTH.S348479

[8] G. Lepetit-Aimon, C. Playout, M. C. Boucher, R. Duval, M. H. Brent, and F. Cheriet, "MAPLES-DR: MESSIDOR anatomical and pathological labels for explainable screening of diabetic retinopathy," *Sci. Data*, vol. 11, no. 1, Aug. 2024. doi: 10.1038/s41597-024-03739-6

[9] W. Liu, H. Lei, H. Xie, B. Zhao, and B. Lei, "Unsupervised domain adaptation based image synthesis and synergistic adversarial learning for optic disc and cup segmentation," in *Proc. 2021 IEEE International Conference on Multimedia and Expo (ICME)*, Shenzhen, China, IEEE, July 2021, pp. 1–6. doi: 10.1109/ICME51207.2021.9428451

[10] S. Wang, L. Yu, X. Yang, C.-W. Fu, and P.-A. Heng, "Patch-based output space adversarial learning for joint optic disc and cup segmentation," *IEEE Trans. Med. Imaging*, vol. 38, no. 11, pp. 2485–2495, Nov. 2019. doi: 10.1109/TMI.2019.2899910

[11] F. Zhang, S. Li, and J. Deng, "Unsupervised domain adaptation with shape constraint and triple attention for joint optic disc and cup segmentation," *Sensors*, vol. 22, no. 22, 8748, Nov. 2022. doi: 10.3390/s22228748

[12] W. Zhou, J. Ji, W. Cui, Y. Wang, and Y. Yi, "Unsupervised domain adaptation fundus image segmentation via multi-scale adaptive adversarial learning," *IEEE J. Biomed. Health Inform.*, pp. 1–12, 2024. doi: 10.1109/JBHI.2023.3342422

[13] Z. Li, M. Liu, Y. Chen, Y. Xu, W. Li, and Q. Du, "Deep cross-domain few-shot learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–18, 2022. doi: 10.1109/TGRS.2021.3057066

[14] R. Selvan, B. Pepin, C. Igel, G. Samuel, and E. B. Dam, "PePR: Performance per resource unit as a metric to promote small-scale deep learning," in *Proc. Northern Lights Deep Learning Conference*, 2025, pp. 220–229.

[15] D. Patel *et al.*, "Traditional machine learning, deep learning, and bert (large language model) approaches for predicting hospitalizations from nurse triage notes: Comparative evaluation of resource management," *JMIR AI*, vol. 3, e52190, Aug. 2024. doi: 10.2196/52190

[16] R. Selvan, J. Schön, and E. B. Dam, "Operating critical machine learning models in resource constrained regimes," in *Proc. 2023 Medical Image Computing and Computer Assisted Intervention*, 2023, pp. 325–335. doi: 10.1007/978-3-031-47425-5_29

[17] Z. Liu *et al.*, "A review of self-supervised, generative, and few-shot deep learning methods for data-limited magnetic resonance imaging segmentation," *NMR in Biomedicine*, vol. 37, no. 8, e5143, Aug. 2024. doi: 10.1002/nbm.5143

[18] S. Li *et al.*, "Few-shot domain adaptation with polymorphic transformers," in *Proc. 2021 Medical Image Computing and Computer Assisted Intervention*, Cham: Springer International Publishing, 2021, pp. 330–340. doi: 10.1007/978-3-030-87196-3_31

[19] 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, Apr. 2022. doi: 10.1049/ipr2.12419

[20] P. H. T. Gama, H. Oliveira, J. A. Dos Santos, and R. M. Cesar, "An overview on meta-learning approaches for few-shot weakly-supervised segmentation," *Computers & Graphics*, vol. 113, pp. 77–88, June 2023. doi: 10.1016/j.cag.2023.05.009

[21] P. A. Zumarsyah, H. A. Nugroho, and I. Ardiyanto, "Few-shot weakly supervised segmentation for retinal fundus images using meta-learning," in *Proc. 2024 5th International Conference on Biomedical Engineering (IBIOMED)*, Bali, Indonesia: IEEE, Oct. 2024, pp. 57–62. doi: 10.1109/iBioMed62485.2024.10875823

[22] P. A. Zumarsyah, I. Ardiyanto, and H. A. Nugroho, "Meta-learners for few-shot weakly-supervised optic disc and cup segmentation on fundus images," *Computers in Biology and Medicine*, vol. 201, 111384, Jan. 2026. doi: 10.1016/j.compbiomed.2025.111384

[23] Z. Chang, Y. Lu, X. Ran, X. Gao, and X. Wang, "Few-shot semantic segmentation: A review on recent approaches," *Neural Comput & Applic*, vol. 35, no. 25, pp. 18251–18275, Sept. 2023. doi: 10.1007/s00521-023-08758-9

[24] P. H. T. Gama, H. Oliveira, J. M. Junior, and J. A. D. Santos, "Weakly supervised few-shot segmentation via meta-learning," *IEEE Trans. Multimedia*, vol. 25, pp. 1784–1797, 2023. doi: 10.1109/TMM.2022.3162951

[25] Y. Han, R. Li, B. Wang, L. Ruan, and Q. Chen, "A pseudo-labeling based weakly supervised segmentation method for few-shot texture images," *Expert Systems with Applications*, vol. 238, 122110, Mar. 2024. doi: 10.1016/j.eswa.2023.122110

[26] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "PANet: Few-shot image semantic segmentation with prototype alignment," in *Proc. 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), Oct. 2019, pp. 9196–9205. doi: 10.1109/ICCV.2019.00929

[27] Y. Wang, N. Luo, and T. Zhang, "Focus on query: Adversarial mining transformer for few-shot segmentation," in *Proc. the 37th International Conference on Neural Information Processing Systems*, May 2024, pp. 31524–31542.

[28] J. Sivaswamy, S. R. Krishnadas, G. Datt Joshi, M. Jain, and A. U. Syed Tabish, "Drishti-GS: Retinal image dataset for Optic Nerve Head (ONH) segmentation," in *Proc. 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, IEEE, Apr. 2014, pp. 53–56. doi: 10.1109/ISBI.2014.6867807

[29] N. Thakur and M. Juneja, "Survey on segmentation and classification approaches of optic cup and optic disc for diagnosis of glaucoma," *Biomedical Signal Processing and Control*, vol. 42, pp. 162–189, Apr. 2018. doi: 10.1016/j.bspc.2018.01.014

[30] R. Kashyap, R. Nair, S. M. P. Gangadharan, M. Botto-Tobar, S. Farooq, and A. Rizwan, "Glaucoma detection and classification using improved u-net deep learning model," *Healthcare*, vol. 10, no. 12, 2497, Dec. 2022. doi: 10.3390/healthcare10122497

[31] S. Sreng, N. Maneerat, K. Hamamoto, and K. Y. Win, "Deep learning for optic disc segmentation and glaucoma diagnosis on retinal images," *Applied Sciences (Switzerland)*, vol. 10, no. 14, July 2020. doi: 10.3390/app10144916

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848

[33] R. Wang, Z. Yang, and Y. Song, "OSAM-Fundus: A training-free, one-shot segmentation framework for optic disc and cup in fundus images," *Biomedical Signal Processing and Control*, vol. 100, 107069, Feb. 2025. doi: 10.1016/j.bspc.2024.107069

[34] Z. Lu and D. Chen, "Weakly supervised and semi-supervised semantic segmentation for optic disc of fundus image," *Symmetry*, vol. 12, no. 1, Jan. 2020. doi: 10.3390/sym12010145

[35] Z. Lu, D. Chen, D. Xue, and S. Zhang, "Weakly supervised semantic segmentation for optic disc of fundus image," *J. Electron. Imag.*, vol. 28, no. 03, May 2019. doi: 10.1117/1.JEI.28.3.033012

[36] Y. Wen *et al.*, "An efficient weakly-supervised learning method for optic disc segmentation," in *Proc. 2020* IEEE *International Conference on Bioinformatics and Biomedicine (BIBM)*, Seoul, Korea (South), Dec. 2020, pp. 835–842. doi: 10.1109/BIBM49941.2020.9313558

[37] Y. Ouassit, S. Ardchir, M. Y. El-Ghoumari, and M. Azouazi, "A brief survey on weakly supervised semantic segmentation," *Int. J. Onl. Eng.*, vol. 18, no. 10, pp. 83–113, July 2022. doi: 10.3991/ijoe.v18i10.31531

[38] K. Zhu, N. N. Xiong, and M. Lu, "A survey of weakly-supervised semantic segmentation," in *Proc. 2023 IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, May 2023, pp. 10–15. doi: 10.1109/BigDataSecurity-HPSC-IDS58521.2023.00013

[39] W. Shen *et al.*, "A survey on label-efficient deep image segmentation: Bridging the gap between weak supervision and dense prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 8, pp. 9284–9305, Aug. 2023. doi: 10.1109/TPAMI.2023.3246102

[40] N. Tajbakhsh, L. Jeyaseelan, Q. Li, J. N. Chiang, Z. Wu, and X. Ding, "Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation," *Medical Image Analysis*, vol. 63, 101693, July 2020. doi: 10.1016/j.media.2020.101693

[41] S. Hu, Z. Liao, and Y. Xia, "Devil is in channels: Contrastive single domain generalization for medical image segmentation," in *Proc. 2023 Medical Image Computing and Computer Assisted Intervention*, 2023, pp. 14–23. doi: 10.1007/978-3-031-43901-8_2

[42] Z. Zhang, Y. Li, and B.-S. Shin, "Robust color medical image segmentation on unseen domain by randomized illumination enhancement," *Computers in Biology and Medicine*, vol. 145, 105427, June 2022. doi: 10.1016/j.compbiomed.2022.105427

[43] X. Bian, C. Wang, W. Liu, and X. Lin, "Unsupervised optic disc segmentation for cross domain fundus image based on structure consistency constraint," in *Proc. 2019 International Conference on Image and Graphics*, 2019, pp. 724–734. doi: 10.1007/978-3-030-34120-6_59

[44] C. Chen and G. Wang, "IOSUDA: An unsupervised domain adaptation with input and output space alignment for joint optic disc and cup segmentation," *Appl. Intell.*, vol. 51, no. 6, pp. 3880–3898, June 2021. doi: 10.1007/s10489-020-01956-1

[45] Y. He, J. Kong, D. Liu, J. Li, and C. Zheng, "Self-ensembling with mask-boundary domain adaptation for optic disc and cup segmentation," *Engineering Applications of Artificial Intelligence*, vol. 129, 107635, Mar. 2024. doi: 10.1016/j.engappai.2023.107635

[46] S. Kadambi, Z. Wang, and E. Xing, "WGAN domain adaptation for the joint optic disc-and-cup segmentation in fundus images," *Int. J. CARS*, vol. 15, no. 7, pp. 1205–1213, July 2020. doi: 10.1007/s11548-020-02144-9

[47] P. Liu, C. T. Tran, B. Kong, and R. Fang, "CADA: Multi-scale collaborative adversarial domain adaptation for unsupervised optic disc and cup segmentation," *Neurocomputing*, vol. 469, pp. 209–220, Jan. 2022. doi: 10.1016/j.neucom.2021.10.076

[48] S.-P. Xu, T.-B. Li, Z.-Q. Zhang, and D. Song, "Minimizing-entropy and fourier consistency network for domain adaptation on optic disc and cup segmentation," *IEEE Access*, vol. 9, pp. 153985–153994, 2021. doi: 10.1109/ACCESS.2021.3128174

[49] P. H. T. Gama, H. Oliveira, and J. A. D. Santos, "Learning to segment medical images from few-shot sparse labels," in *Proc. 2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, Oct. 2021, pp. 89–96. doi: 10.1109/SIBGRAPI54419.2021.00021

[50] C. Lang, G. Cheng, B. Tu, C. Li, and J. Han, "Base and meta: A new perspective on few-shot segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10669–10686, Sept. 2023. doi: 10.1109/TPAMI.2023.3265865

[51] G. Cheng, C. Lang, and J. Han, "Holistic prototype activation for few-shot segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4650–4666, Apr. 2023. doi: 10.1109/TPAMI.2022.3193587

[52] J. I. Orlando *et al.*, "REFUGE challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs," *Medical Image Analysis*, vol. 59, 101570, Jan. 2020. doi: 10.1016/j.media.2019.101570

[53] F. Fumero, J. Sigut, S. Alayón, M. González-Hernández, and M. González, "Interactive tool and database for optic disc and cup segmentation of stereo and monocular retinal fundus images," in *Proc. WSCG 2015 Conference on Computer Graphics, Visualization and Computer Vision*, 2015.

[54] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012. doi: 10.1109/TPAMI.2012.120

[55] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. New York, New York: Pearson Education, 2018.

[56] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Computer Vision–ECCV 2018*, 2018, pp. 833–851. doi: 10.1007/978-3-030-01234-2_49

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, June 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90

[58] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, June 2018, pp. 4510–4520. doi: 10.1109/CVPR.2018.00474

[59] J. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021. doi: 10.1109/TPAMI.2020.2983686

[60] D. Kingma and J. Ba, "Adam: a method for stochastic optimization," in *Proc. International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

[61] A. A. Taha, A. Hanbury, and O. A. J. del Toro, "A formal method for selecting evaluation metrics for image segmentation," in *Proc. 2014 IEEE International Conference on Image Processing*, IEEE, Jan. 2014, pp. 932–936. doi: 10.1109/ICIP.2014.7025187

[62] L. Maier-Hein *et al.*, "Metrics reloaded: Recommendations for image analysis validation," *Nat. Methods*, vol. 21, no. 2, pp. 195–212, Feb. 2024. doi: 10.1038/s41592-023-02151-z

[63] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. the 24th International Conference on Neural Information Processing Systems*, 2011, pp. 2546–2554.

[64] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6765–6816, Jan. 2017.

[65] Y. Ozaki, M. Nomura, and M. Onishi, "Hyperparameter optimization methods: Overview and characteristics," *IEICE Trans. Inf. Syst.*, no. 9, pp. 615–631, Sept. 2020. doi: 10.14923/transinfj.2019JDR0003

[66] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA: Curran Associates Inc., 2019, pp. 8026–8037.

[67] W. Falcon *et al.*, *PyTorchLightning/Pytorch-Lightning: 0.7.6 Release*, 2020. doi: 10.5281/ZENODO.3828935

[68] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sept. 2020. doi: 10.1038/s41586-020-2649-2

[69] S. van der Walt *et al.*, "Scikit-image: Image processing in Python," *PeerJ*, vol. 2, e453, June 2014. doi: 10.7717/peerj.453

[70] P. Liu, B. Kong, Z. Li, S. Zhang, and R. Fang, "CFEA: Collaborative Feature ensembling adaptation for domain adaptation in unsupervised optic disc and cup segmentation," in *Proc. 2019 Medical Image Computing and Computer Assisted Intervention*, 2019, pp. 521–529. doi: 10.1007/978-3-030-32254-0_58

[71] C. Chen, Q. Dou, H. Chen, J. Qin, and P. A. Heng, "Unsupervised bidirectional cross-modality adaptation via deeply synergistic image and feature alignment for medical image segmentation,"

IEEE Trans. Med. Imaging, vol. 39, no. 7, pp. 2494–2505, July 2020. doi: 10.1109/TMI.2020.2972701

[72] L. Liu, Z. Zhang, S. Li, K. Ma, and Y. Zheng, "S-CUDA: Self-cleansing unsupervised domain adaptation for medical image segmentation," *Medical Image Analysis*, vol. 74, 102214, Dec. 2021. doi: 10.1016/j.media.2021.102214

[73] F. Wu and X. Zhuang, "Minimizing estimated risks on unlabeled data: A new formulation for semi-supervised medical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–17, 2022. doi: 10.1109/TPAMI.2022.3215186