

DLTTA: Dynamic Learning Rate for Test-Time Adaptation on Cross-Domain Medical Images

Hongzheng Yang, Cheng Chen¹, *Member, IEEE*, Meirui Jiang², *Student Member, IEEE*,
Quande Liu², *Student Member, IEEE*, Jianfeng Cao, Pheng Ann Heng³, *Senior Member, IEEE*,
and Qi Dou⁴, *Member, IEEE*

Abstract—Test-time adaptation (TTA) has increasingly been an important topic to efficiently tackle the cross-domain distribution shift at test time for medical images from different institutions. Previous TTA methods have a common limitation of using a fixed learning rate for all the test samples. Such a practice would be sub-optimal for TTA, because test data may arrive sequentially therefore the scale of distribution shift would change frequently. To address this problem, we propose a novel dynamic learning rate adjustment method for test-time adaptation, called *DLTTA*, which dynamically modulates the amount of weights update for each test image to account for the differences in their distribution shift. Specifically, our *DLTTA* is equipped with a memory bank based estimation scheme to effectively measure the discrepancy of a given test sample. Based on this estimated discrepancy, a dynamic learning rate adjustment strategy is then developed to achieve a suitable degree of adaptation for each test sample. The effectiveness and general applicability of our *DLTTA* is extensively demonstrated on three tasks including retinal optical coherence tomography (OCT) segmentation, histopathological image classification, and prostate 3D MRI segmentation. Our method achieves effective and fast test-time adaptation with consistent performance improvement over current state-of-the-art test-time adaptation methods. Code is available at <https://github.com/med-air/DLTTA>.

Index Terms—Test-time adaptation, cross-domain medical image analysis, distribution shift, dynamic learning rate.

I. INTRODUCTION

DESPITE recent progress on domain adaptation techniques [1]–[4], deep learning models remain difficult to generalize across medical datasets with heterogeneous data

Manuscript received 20 May 2022; accepted 7 July 2022. Date of publication 15 July 2022; date of current version 2 December 2022. This work was supported in part by the Hong Kong Innovation and Technology Fund under Project ITS/238/21, Project ITS/170/20, and Project GHP/110/19SZ, in part by The Chinese University of Hong Kong Shun Hing Institute of Advanced Engineering under Project MMT-p5-20, and in part by the Shenzhen-Hong Kong Collaboration Development Zone. (Corresponding author: Cheng Chen.)

Hongzheng Yang is with the Department of Artificial Intelligence, Beihang University, Beijing 100191, China (e-mail: yanghongz@buaa.edu.cn).

Cheng Chen, Meirui Jiang, Quande Liu, Jianfeng Cao, Pheng Ann Heng, and Qi Dou are with the Department of Computer Science and Engineering and the Institute of Medical Intelligence and XR, The Chinese University of Hong Kong, Hong Kong, SAR (e-mail: cchen@cse.cuhk.edu.hk; mrjiang@cse.cuhk.edu.hk; qdliu@cse.cuhk.edu.hk; jfcao@cse.cuhk.edu.hk; pheng@cse.cuhk.edu.hk; qdou@cse.cuhk.edu.hk).

Digital Object Identifier 10.1109/TMI.2022.3191535

distributions, that are caused by varying image acquisition conditions (e.g., imaging protocols and scanners) [5], [6]. This problem severely hinders the deployment of established deep learning models to new test samples with unknown data distributions. Existing unsupervised domain adaptation (UDA) methods [7]–[15] typically require to assemble a large test dataset to use concurrently with the training data to conduct distribution alignment for improved prediction performance on test samples. However, this setting is still problematic for real-world model deployment due to two important reasons. Firstly, it is not efficient, if realistic, to wait for the accumulation of sufficient amount of test samples (e.g., thousands of instances), since the test data usually arrive sequentially one by one, or batch by batch. However, immediate prediction on a single test sample or a batch of instances is highly desired in clinical practice for timely diagnosis and treatment for patients. Secondly, accessing the training dataset at test time is practically difficult, because data sharing across hospitals is prohibitive due to the privacy concern for medical data. Regarding these limitations, it would be more convenient if a given trained model could be quickly and continuously adapted to each test sample at inference time, without using any training data.

This motivates the topic of test-time adaptation (TTA), and a few promising early investigations have been very recently conducted [16]–[20]. The general idea is to gradually adjust model parameters by exploiting the distributional information provided by each test sample. A representative work is test-time training (TTT) [16] which adds an auxiliary branch with self-supervision of rotation prediction to adapt the model encoder with the test data. Sharing similar idea, the method DTTA [20] and ATTA [21] employ autoencoders to learn shape priors and to align feature space respectively, achieving promising results on cross-domain medical image analysis at test time. The latest state-of-the-art fully test-time adaptation method is Tent [17] which proposes to adapt the batch normalization (BN) layer by minimizing the entropy of model predictions on test data.

In this paper, we identify the common limitation in current test-time adaptation literature, i.e., applying the same learning rate for all the test samples. This existing practice is sub-optimal since the sequentially arriving test data may not have the same distribution shift, hence can differ largely in their adaptation demand. For example, as shown in Fig. 1, the test samples can present apparently varying degree of

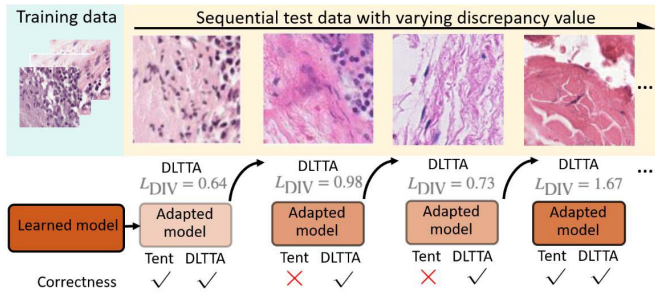


Fig. 1. Illustration of the varying distribution shift of test data and our dynamic learning rate adjustment strategy for test-time adaptation (DLTTA). Test images even from the same dataset present large appearance variations compared to the training data. Our estimated discrepancy values L_{DIV} also indicate the distribution shift variations. By adapting the model dynamically, our DLTTA obtains more accurate predictions than the state-of-the-art test-time adaptation method Tent.

appearance changes, even though they come from the same test dataset (e.g., images stained in one hospital). This matters for test-time adaptation of the learned models. Intuitively, for test data with severe distribution shift, the model requires greater update; while for test data with mild distribution shift in comparison to training statistics, the model should not be adapted too dramatically, otherwise might drift away from the discriminative features learned from the massive training data. Moreover, for test-time adaptation, the model is continuously updated. Therefore, even for two test samples with similar distribution shift but arrive at different time points, the demanded degree of adaptation would also be different due to the transitions in model parameters over time. To achieve such an adaptation demand, we argue it is important to dynamically adjust the test-time learning rate for each test sample, as learning rate is core to control the scale of updating the model weights in response to the estimated TTA loss during optimization.

With above insights, we propose a novel dynamic learning rate method for test-time adaptation (DLTTA). The goal is to adapt the model dynamically to account for the differences in adaptation demand across test samples that arrive sequentially. Firstly, to capture the progressive change of the model, we maintain a memory bank to cache the feature and prediction pairs of previous test data in a dynamic manner. Then, to estimate the discrepancy of a newly coming test sample, we retrieve semantically similar ones from the memory bank and calculate the Kullback–Leibler (KL) divergence between the cached predictions and the current output for the derived discrepancy. Based on the estimated discrepancy, we further adjust the learning rate to achieve suitable adaptation for online test data. Compared with previous methods, the dependence on the choice of initial learning rate can be substantially alleviated owing to our adaptive learning rate strategy. Importantly, our method is designed to be simple yet effective, so that it can be flexibly incorporated to different existing TTA frameworks to improve adaptation performance, as well as be generally applied to different medical image analysis tasks with various model architectures. Our main contributions are as follows:

- We propose a new dynamic learning rate adjustment framework for test-time adaptation on cross-domain medical image analysis. To the best of our knowledge, this is the first work of its kind to explore a dynamic learning strategy to overcome the varying distribution shift of inference data for model adaptation at test time.
- We devise a novel memory bank-based discrepancy estimation strategy to model the adaptation demand, based on which the learning rate is specifically adjusted for each particular test sample in an online manner.
- We have validated the effectiveness of our method for both classification and segmentation tasks on three different medical imaging modalities with 2D or 3D models. Experimental results show that our DLTTA is generic to different network architectures and can consistently outperform current state-of-the-art TTA methods.

II. RELATED WORKS

A. Test-Time Adaptation

Adapting deep models solely based on unlabeled test data that arrive sequentially has recently drawn increasing interests. Unlike conventional UDA methods that require to access the training data and assemble sufficient amount of test data, the test-time adaptation methods are able to update a model with the distributional information provided by a single or a batch of test data [16]–[18], [22]–[24]. Prediction-time batch normalization (PTBN) [18] re-estimates BN statistics [25] based on the test data, which is simple but effective. TTT [16] adapts the feature extractor at test time by leveraging an auxiliary self-supervised task of rotation prediction. TTT++ [22] improves TTT by further aligning the first- and second-order statistics of the training and test data and adopts the self-supervised task of contrastive learning [26]. Tent [17] proposes to adapt the affine parameters in BN layers at test time by minimizing entropy of model predictions. T3A [23] adjusts the classifier of a trained source model by computing a pseudo-prototype representation of different classes using unlabeled test data.

Tackling cross-domain distribution shift under more realistic settings has also been recently attempted in medical image applications [10], [13], [14], [19]–[21], [27], [28]. Test-time UDA [10] performs adversarial learning on each test sample separately to align the distribution with the source training data. Liu *et al.* [29] propose a generative self-training test-time UDA framework for cross-domain magnetic resonance imaging synthesis. Zhu *et al.* [28] design test-time training on each test image pair to improve the generalization of learning-based registration. Wang *et al.* [30] make a model adaptive to a specific test image by bounding box and scribble-based interactive fine-tuning. To achieve test-time adaptation, DTTA [20] trains denoising autoencoders in the source domain to learn shape priors for adaptation at test time. Similarly, Valvano *et al.* [31] keep mask discriminators to provide shape prior and fine-tune the segmentor on each individual test instance. The latest method ATTA [21] trains a set of autoencoders on the source dataset and updates a set of adaptors at test time with autoencoders' reconstruction

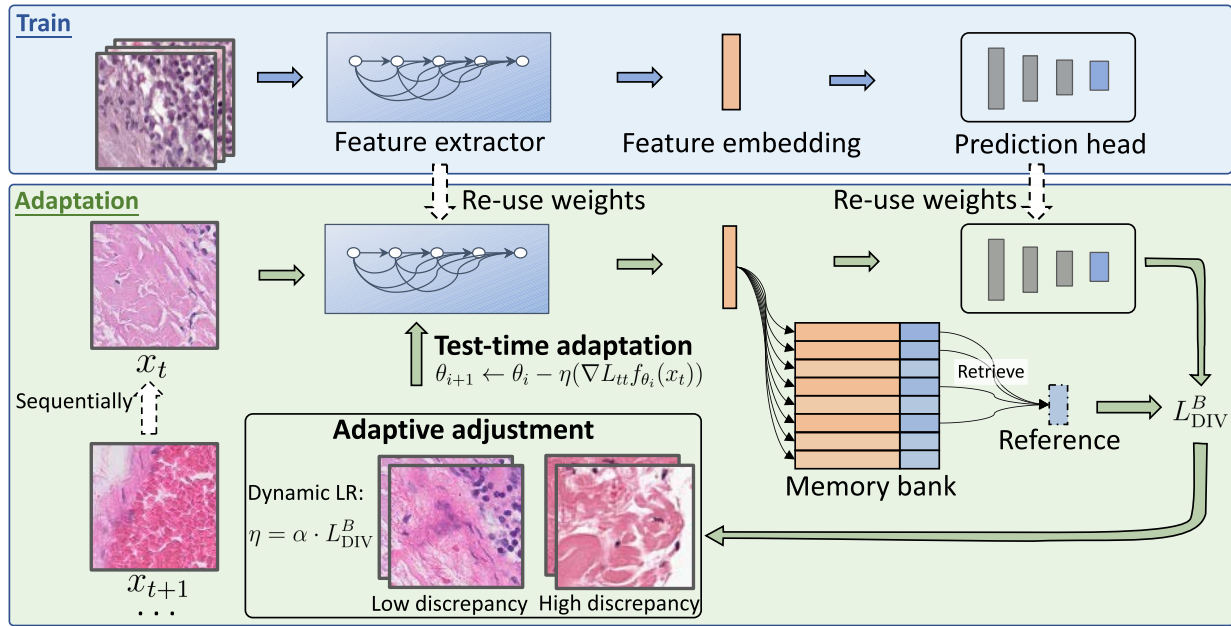


Fig. 2. Overview of our novel method for dynamic learning rate adjustment of test-time adaptation (DLTTA), to deal with the varying distribution shift of sequentially coming test data. A memory bank is constructed to derive the discrepancy L_{DIV}^B between the current predictions and the semantically similar reference retrieved and calculated from the memory bank, and the test-time learning rate is then adaptively adjusted based on the estimated discrepancy to achieve desired adaptation for each test sample.

loss, showing more efficient inference and better results than previous works.

Existing test-time adaptation methods mainly focus on designing different TTA losses, but adopt the same step size for adaptation on all test samples. The strategy of a fixed step size is sub-optimal since for real-world data streams, the test data distributions may change frequently in a non-stationary way [32]. When adapting a model to new environment at test time, we consider dynamic learning rate adjustment is important to control the amount of weights update at each iteration for effective and efficient model adaptation.

B. Dynamic Learning Rate

To adjust the learning rate during a model training process, some predefined learning rate schedulers, such as linear step decay [33], exponential decay, cosine/sine annealing [34], have been proposed and widely used in training deep neural networks. However, naively decaying or cycling the learning rate might be insufficient for complex situations, such as the test-time adaptation when facing the new environment with varying distribution shift. There are also works that exploit gradients for learning rate adjusting. Maclaurin *et al.* [35] introduce a reversible learning technique to compute gradients with respect to the learning hyperparameters. The learning rate is then adjusted through an inner optimization. Baydin *et al.* [36] compute the gradient with respect to the learning rate and dynamically adjusts the learning rate updates in an online manner at each iteration. Although these methods can dynamically adjust the learning rate by gradient update, accurate supervision from labeled data is needed to calculate the update gradient signals, which is not available in test-time adaptation scenarios.

Our insight is to achieve dynamic learning rate adjustment according to a estimated distribution shift. How to measure the discrepancy solely based on the model parameters and the current test sample in an unsupervised way is challenging and remains unsolved yet. A recent work [37] which requires source training data for adaptation proposes to infer “easier” and “harder” test data by calculating a confidence score based on the output consistency of multiple classifiers. Although this work shares similar idea to ours on estimating the discrepancy of test images, it restricts to specific network architectures with multiple different classifiers and it uses the identified “easier” test data to produce pseudo labels instead of designing a dynamic adaptation strategy. Lee *et al.* [38] exploit the stored BN statistics of training data to compare with the test data as a measure of distribution shift. Due to the continuous update of the model parameters at test time, the BN statistics of training data fail to represent the updated model status, leading to inaccurate discrepancy estimation. Our proposed memory bank-based discrepancy measurement captures both the model progress and the distribution variation of test data, thus can provide more up-to-date discrepancy estimation for effective dynamic learning rate adjustment of test-time adaptation.

III. METHODS

In this section, we first provide the overall framework of test-time adaptation. We then introduce our method for dynamic learning rate of test-time adaptation, in which a memory bank is incorporated to estimate the discrepancy of predictions at inference stage. With the estimated discrepancy, a dynamic learning rate is devised to dynamically modulate the test-time adaptation process for stable and improved adaptation results. An overview of our framework for effective test-time adaptation is illustrated in Fig. 2.

A. Test-Time Adaptation Overall Framework

When applying a model to new test data, a problem is the new test samples may follow an unknown data distribution, leading to severe performance drop of model prediction. Previous domain adaptation methods require to assemble a sufficient number of test data, while in real-world applications, the test sample usually arrives sequentially with varying distribution shift. The more appealing solution, test-time adaptation, aims to continuously adapt the model directly according to each presented test sample. Specifically, for test-time adaptation, we are only given a pre-trained model f_{θ^s} parameterized by a set of parameters θ^s and new test samples $\{x_1, x_2, \dots, x_{t-1}, x_t\}$ that arrive online sequentially with varying distribution shift. The model f_{θ^s} is optimized with source training data $\{(x_i^s, y_i^s)\}_{i=1}^{N^s}$ by empirical risk minimization as below:

$$\theta^s = \arg \min_{\theta} \frac{1}{N^s} \sum_{i=1}^{N^s} L_m(f_{\theta}(x_i^s), y_i^s), \quad (1)$$

where L_m denotes the supervised loss for training, such as the cross entropy loss. The model f_{θ^s} would perform poorly on new test sample x_t that arrives at time step t and differs from the source training samples in data distributions. Then to continuously adapt the model to achieve better generalization performance, a test-time objective function L_{tt} needs to be dedicatedly designed to update the model parameters based on the distributional information provided by each test sample. For a test sample x_t appearing at the t -th iteration of model update, we have:

$$\theta_{t+1} \leftarrow \theta_t - \eta(\nabla L_{tt} f_{\theta_t}(x_t)), \quad (2)$$

where η denotes the learning rate for test-time adaptation, θ_1 is initialized with θ^s . The updated model $f_{\theta_{t+1}}$ is used to obtain prediction of test sample x_t . Different test-time objective function L_{tt} has been proposed in prior works, such as rotation prediction loss [16], entropy minimization of model predictions [17], and autoencoder reconstruction loss [21].

B. Dynamic Learning Rate on Test Data

Previous test-time adaptation methods adopt a fixed learning rate η in Eq. (2), which needs to be carefully chosen since the adaptation performance is sensitive to the learning rate [17]. We argue that a static learning rate for all adaptation steps cannot accurately update the model to overcome the varying distribution shift of test data. We therefore propose a dynamic learning strategy to capture both the model changes and different shift degree of test data during test-time adaptation.

1) Memory Bank Construction for Discrepancy Estimation: For dynamic adaptation, it is critical to know the demanded adaptation extent at each update step, which is the key factor of adjusting the learning rate. To measure the extent of adaptation, we propose a memory bank-based discrepancy estimation. The memory bank is constructed to store the latest pairs of feature representation and prediction mask extracted by the continuously updated model. Such pairs reflect the model's change, and can be further utilized to calculate the

distance with incoming test samples to estimate the distribution shift degree. With the shift degree measurement, we are able to adjust the test-time learning rate accordingly.

Specifically, the memory bank \mathcal{M} comprises of K pairs of keys and values $\{(q_k, v_k)\}_{k=1}^K$. As shown in Fig. 2, the keys $\{q_k | q_k = h(x_t)\}$ are feature maps computed by the feature extractor h of a model and the values $\{v_k | v_k = g(q_k)\}$ correspond to the prediction masks generated by the classifier head g . We update \mathcal{M} by caching new (q_k, v_k) pairs continuously. Since the model is progressively updated at test time, the early elements in the memory bank cannot indicate the most recent model performance on test data. We thus only maintain the memory bank with a fixed size K and holds the First In First Out (FIFO) principle when writing new pairs. Then to measure the discrepancy of a newly coming test sample x_t , we aim to retrieve a support set $\mathcal{R} = \{(q_d, v_d)\}_{d=1}^D \subseteq \mathcal{M}$ from the memory bank, containing features and predictions of elements that are semantically similar to the current test sample x_t . This is achieved by computing the D -nearest neighbors based on the L2 distance between each key and the features of the query sample $h(x_t)$. Since the key maintains semantic-level contextual features, the support set \mathcal{R} retains internal agreement regarding high-level information, e.g., object categories in an image. The ensembling of predictions in \mathcal{R} can be used as the reference prediction for x_t as:

$$\hat{p}_t = \frac{1}{D} \sum_{d=1}^D v_d. \quad (3)$$

Then the prediction discrepancy of image x_t is derived by:

$$L_{\text{DIV}} = \frac{1}{2} (L_{\text{KL}}(\hat{p}_t || f_{\theta_t}(x_t)) + L_{\text{KL}}(f_{\theta_t}(x_t) || \hat{p}_t)),$$

with

$$L_{\text{KL}}(\hat{p}_t || f_{\theta_t}(x_t)) = \sum_j \hat{p}_t(j) \log \frac{\hat{p}_t(j)}{f_{\theta_t}(x_t(j))}, \quad (4)$$

where L_{KL} denotes the KL divergence and $f_{\theta_t}(x_t)$ denotes the prediction of x_t generated by the model f_{θ_t} .

2) Adaptive Learning Rate Adjustment: With the discrepancy measured with Eq. (4), our intuition is that high discrepancy indicates a significant gap that needs to be largely bridged and low discrepancy requires smaller adaptation. We therefore propose to dynamically adjust the self-supervised learning rate based on the estimated discrepancy. To describe our method in a more general way, we adopt a batch-wise formulation to account for that the test samples may arrive one by one or batch by batch. A single test image corresponds to the batch size 1. Given a batch of test data $\{x_{t,b}\}_{b=1}^B$ at each test-time adaptation step, we calculate the total discrepancy by:

$$L_{\text{DIV}}^B = \frac{1}{B} \sum_{b=1}^B L_{\text{DIV},b}, \quad (5)$$

where B is the batch size. L_{DIV}^B captures the general discrepancy distributed over batched predictions. Thereafter, batch-wise dynamic learning rate is obtained through a function

$H(L_{\text{DIV}}^B)$ which outputs learning rate for test-time adaptation task directly based on the integral discrepancy L_{DIV}^B :

$$H(L_{\text{DIV}}^B) = \alpha \cdot L_{\text{DIV}}^B, \quad (6)$$

where α scales the learning rate further and could be empirically set as the value used for the model optimization with the source training images.

Our proposed dynamic learning rate adjustment at test time can be easily deployed to any network architecture and self-supervised objective function to improve the test-time adaptation process. Since the feature representation and prediction masks are naturally calculated and stored in the memory bank, the computational cost at test time mainly comes from the retrieval of support set for the calculation of reference prediction \hat{p}_t . We improve the efficiency by keeping a small retrieval size (e.g. 8). The effect of retrieval size is analyzed in the experiment and results show that a relatively small size can already achieve good results, increasing the size (e.g. increasing to 20) does not show higher performance.

C. Learning Process and Implementation Details

1) Learning Process: The overall procedure to perform our test-time adaptation is summarized as follows. The model to be adapted is initialized with parameters learned from the source training data. For a coming test sample, the model firstly performs a forward pass to obtain the pair of feature representation and prediction mask to retrieve semantically similar elements from the memory bank, and calculates the discrepancy and dynamic learning rate for the specific test sample. The model is then updated by using the derived learning rate with a test-time objective function to achieve desired adaptation. Note that only a partial model parameters are adapted following previous works with different test-time adaptation objectives [16], [17], [21]. After the one-step adaptation, the model performs another forward pass to obtain the prediction for the current test sample and update the new element pair for the memory bank. This model is iteratively updated with the previous process with each sequentially coming test sample. Before the memory bank being constructed, the learning rate is kept as the initial value. The pseudo-code of DLTTA is shown in Algorithm 1.

2) Implementation Details: The choice of network architecture in our method is flexible. Without loss of generality, we followed [21] to employ a U-Net [39] for retinal OCT image segmentation, followed [40] to use a DenseNet-121 [41] pre-trained on ImageNet for histopathological image classification, and employed a 3D U-Net [42] for prostate MRI image segmentation. The encoders of our 2D U-Net and 3D U-Net contain 4 Convolution-BatchNorm-ReLU blocks, which continuously down-sample the image resolution and double the feature channels dimensions. Then the features are processed by a bottleneck layer and fed into a decoder, which has 4 transpose convolution blocks to up-sample the intermediate feature maps. All 2D convolutions use kernel size 3×3 and 3D convolutions use kernel size $3 \times 3 \times 3$. The encoder features are skip-connected to the decoder at each stage and the feature channel dimensions are [16, 32, 64, 128] for each

Algorithm 1 Dynamic Learning Rate for Test-Time Adaptation (DLTTA) Learning Process

Input: Test samples or batches $\{x_{t,b}\}$ for t from 1 to T and b from 1 to B , model parameters learned from the source training data θ_1 , memory bank \mathcal{M} , learning rate for source train images α , test-time batch size B

Output: The final prediction $\{\bar{y}_t\}_{t=1}^T$ for test samples

```

1: Initialize model  $f = g \circ h$  with parameters  $\theta_1$ 
2: for  $t = 1, 2, \dots, T$  do
3:   for  $b = 1, 2, \dots, B$  do
4:      $q_{t,b} = h(x_{t,b}), v_{t,b} = g(q_{t,b})$   $\triangleright$  Forward pass using  $\theta_t$ 
5:      $\{(q_d, v_d)\}_{d=1}^D \sim \mathcal{M}$   $\triangleright$  Retrieve support set
6:      $\hat{p}_{t,b} = \frac{1}{D} \sum_{d=1}^D v_d$   $\triangleright$  Eq. (3)
7:      $L_{\text{DIV},b} = \frac{1}{2} (L_{\text{KL}}(\hat{p}_{t,b} || f_{\theta_t}(x_{t,b})) + L_{\text{KL}}(f_{\theta_t}(x_{t,b}) || \hat{p}_{t,b}))$   $\triangleright$  Eq. (4)
8:   end for
9:    $L_{\text{DIV}}^B = \frac{1}{B} \sum_{b=1}^B L_{\text{DIV},b}$   $\triangleright$  Eq. (5)
10:   $\eta = \alpha \cdot L_{\text{DIV}}^B$   $\triangleright$  Eq. (6)
11:   $\theta_{t+1} \leftarrow \theta_t - \eta (\nabla L_{tt} f_{\theta_t}(x_t))$   $\triangleright$  Dynamic learning
12:   $(q_t, v_t), \bar{y}_t = f_{\theta_{t+1}}(x_t)$   $\triangleright$  Forward pass using  $\theta_{t+1}$ 
13:   $\mathcal{M}.\text{add}((q_t, v_t))$   $\triangleright$  Update the memory bank
14: end for
15: return  $\{\bar{y}_t\}_{t=1}^T$ 

```

block. For segmentation network, we consider the encoder and bottleneck layer as feature extractor and the decoder as the prediction head. The architecture of DenseNet-121 follows the implementation in Torchvision [43] library, which consists of 4 densely connected blocks followed by global pooling and fully connected classification layers. For each task, the network backbone of our method and other comparison approaches are the same to ensure fair comparison.

For model training with the source images, model was trained from scratch for 100 epochs with Adam optimizer and learning rates were initialized as $1e-3$ and $3e-4$ for segmentation and classification tasks. For test-time adaptation of our method and all the comparison baselines, we performed one-step adaption for each batch of test data with batch size 1 for segmentation and 200 for classification task. We stored key and value pairs for the latest $\frac{K}{B}$ adaptation steps. We empirically set $\frac{K}{B}$ as 20 for segmentation and 4 for classification. The retrieval size D is set to 8 for segmentation task and 12 for classification task. Following [18], BN statistics were re-collected from test data for all the methods. The framework was implemented with Pytorch 1.7.0, and trained on one NVIDIA TitanXp GPU.

IV. EXPERIMENTS

A. Datasets and Evaluation Metrics

We first validate the effectiveness of our method on retinal layer segmentation with optical coherence tomography (OCT) datasets that present distribution shift caused by the different Heidelberg Spectralis scanner and Cirrus scanner [21], [47]. Then we evaluate our method on a much larger dataset of histopathological images for tumor and normal tissue classification, that are obtained from different hospitals with varying

protocols and patient populations [48]–[50]. We also validate our method on prostate segmentation with 3D MRI images collected from different medical centers [51]–[54].

1) Retinal OCT Image Segmentation: We follow the literature [21] to employ a public OCT dataset [47] that is acquired from Heidelberg Spectralis scanners as the source training data, and a public dataset of OCT images obtained from Cirrus scanners [21] as the shifted test dataset. The source training dataset consists of 35 3D volumes, but due to the large physical distance between slices, following [47], the 3D volumes were split into 588 2D slices for training, 147 slices for validation, and 980 slices for testing. The task is to segment eight different retinal layers with ground truth being provided for model training. The test dataset contains 48 slices from 6 subjects. As in [21], the Cirrus 2D slices were re-sampled to have the same physical within-slice resolution as the Spectralis scans. All the slices from both datasets were pre-processed with retina flattening and cropped and resized to 128×1024 . For evaluation, we follow previous works [21] of using the same two OCT datasets to adopt the popular Dice score to evaluate the segmentation performance.

2) Histopathological Image Classification: We use the large Camelyon17 [48]–[50] dataset to evaluate our method, which contains more than 400,000 histopathological image patches of size 96×96 from five different hospitals. In our experiment, 302,436 tissue patches collected from three hospitals are used as the source training data, 34,904 patches and 85,054 patches from two other hospitals are taken as two unseen test datasets respectively, i.e., unseen dataset A and unseen dataset B. The task is to predict whether a given region of tissue contains any tumor tissue. Following [48], we normalized each image patch to zero mean and unit variance with the data statistics of ImageNet. Five most commonly used evaluation metrics of classification task, including accuracy, sensitivity, specificity, AUC, and F1 score are adopted to evaluate different methods.

3) Prostate MRI Image Segmentation: We further validate our approach on the prostate segmentation task across multi-center 3D MRI data. We use prostate T2-weighted MRI volumes of three public datasets NCI-ISBI13 [52], I2CVB [54], and PROMISE12 [53], which were collected from six clinical centers. We follow the previous work [51] to partition the data into six datasets A to F, according to the clinical centers that the datasets were collected from. Dataset A is employed as the training data and the other datasets are used as different unseen test datasets. For data pre-processing, each MRI volume was normalized to zero mean and unit variance in intensity values and resized to 384×384 in the axial plane. Patches with size of $80 \times 80 \times 80$ were cropped as the network inputs. The common evaluation metric Dice score is adopted to compare the performance of different methods.

B. Comparison With State-of-the-Art Methods

Our dynamic learning strategy can be deployed to different test-time adaptation methods to improve the effectiveness of model update. For each task, we deploy our dynamic learning rate strategy to three representative test-time adaptation methods, including **TTT** [16] which adapts the model with

a proxy task of rotation prediction, **Tent** [17] which adjusts the batch normalization layers by minimizing the entropy of model predictions on test data, and **ATTA** [21] which reduces domain shift with the autoencoders' reconstruction loss. We also compare our method with **PTBN** [18] which re-estimates the batch normalization statistics from the test data. To provide strong baselines, we include UDA methods in comparison that concurrently utilize source training data and the entire test dataset for adaptation. For retinal OCT image segmentation, we include three UDA methods that have been employed to adapt OCT images in [21], that are the image translation methods **UDA-ST** [44] and **CycleGAN** [45], and the widely used output space adaptation method **UDAS** [46]. For histopathological image classification, we compare with **UDA-SwAV** [40], which is the best reported UDA model for histopathological image classification using the datasets as ours. For prostate segmentation, we also employ **UDAS** for comparison. The results of ATTA, UDA-ST on OCT images and the results of UDA-SwAV on unseen histopathological dataset A are directly referenced from [21] and [40] respectively since the same datasets, network backbones, and data split are used in their methods and ours. The other results are obtained by re-implementing based on the released code with the network backbone being consistent for all the comparison methods. In addition, the **W/o Adaptation** model denotes directly applying the model optimized with training dataset to obtain the predictions of test samples.

1) Results for Retinal Layer Segmentation: Table I presents the comparison results for retinal layer segmentation from OCT test dataset of Cirrus images. We can see that all test-time adaptation and UDA methods improve over the "W/o Adaptation" baseline, showing the benefits of model adaptation towards the data distribution of test samples. Our DLTTA equipped with the dynamic learning rate strategy consistently improves different test-time adaptation methods TTT, Tent and ATTA for the segmentation of all retinal layers. These results demonstrate the effectiveness of our dynamic learning rate on improving the test-time model update to overcome the varying distribution shift of test data, and validate that our dynamic learning rate strategy can be applied to different test-time adaptation methods to improve adaptation performance. With deploying to ATTA, our method achieves the highest mean Dice values 83.41%, which outperforms all UDA methods utilizing the source training data during adaptation process. Plain Tent obtains comparable performance to UDA-ST, but further adding our dynamic learning rate strategy improves the mean Dice from 81.50% to 82.63%, outperforming UDA-ST. UDAS outperforms plain Tent but performs slightly worse than ATTA. When ATTA is equipped with the dynamic learning rate strategy, it further outperforms UDAS. This shows the great potential of dynamic test-time adaptation and indicates that the source training data are not necessarily needed to achieve good adaptation performance. The possible reason could be that the dynamic test-time adaptation updates the model directly based on the distributional information of each specific test sample, instead of trying to find a domain-invariant space for the entire training and test data which could be difficult when the training and test images differ significantly in data distributions.

TABLE I

COMPARISON OF DIFFERENT METHODS IN RETINAL LAYER SEGMENTATION IN DICE SCORE ON OCT TEST DATASET OF CIRRUS IMAGES

Method	RNFL	GCIP	INL	OPL	ONL	IS	OS	RPE	Average
W/o Adaptation	64.61±0.08	68.88±0.02	62.34±0.06	56.64±0.03	83.03±0.02	85.51±0.03	88.27±0.03	82.85±0.03	74.01±0.05
PTBN [18]	67.53±0.19	80.29±0.19	72.56±0.17	61.93±0.09	85.42±0.14	86.17±0.06	88.94±0.06	84.03±0.04	78.36±0.09
UDA-ST [44]	77.30±0.09	86.60±0.05	78.20±0.05	64.40±0.06	87.60±0.03	88.30±0.03	86.90±0.05	82.60±0.05	81.50±0.05
CycleGAN [45]	77.23±0.03	86.49±0.10	77.83±0.05	63.88±0.10	86.44±0.03	88.27±0.01	86.64±0.05	82.44±0.04	81.15±0.06
UDAS [46]	78.62±0.14	87.04±0.13	79.23±0.15	65.09±0.08	88.40±0.09	88.58±0.07	86.42±0.11	83.89±0.24	82.16±0.12
TTT [16]	70.55±0.18	81.83±0.12	74.14±0.16	64.41±0.08	86.76±0.09	86.85±0.12	89.15±0.08	84.27±0.08	79.75±0.16
DLTTA (<i>Ours</i>)	71.09±0.14	82.97±0.12	75.30±0.18	65.36±0.14	87.05±0.11	88.31±0.06	89.39±0.09	84.87±0.05	80.54±0.11
Tent [17]	76.95±0.13	83.93±0.08	75.49±0.16	66.76±0.13	88.10±0.12	87.08±0.15	89.28±0.13	84.38±0.08	81.50±0.15
DLTTA (<i>Ours</i>)	77.60±0.06	84.27±0.04	75.78±0.07	69.08±0.05	90.38±0.06	88.42±0.09	90.01±0.03	85.61±0.10	82.63±0.06
ATTA [21]	73.20±0.13	85.10±0.04	79.10±0.03	69.20±0.06	90.40±0.02	88.40±0.03	89.90±0.02	84.90±0.03	82.50±0.05
DLTTA (<i>Ours</i>)	73.86±0.08	86.48±0.09	79.86±0.04	70.18±0.02	91.57±0.07	89.69±0.04	90.12±0.05	85.56±0.03	83.41±0.05

RNFL: Retina nerve fiber layer, GCIP: Ganglion cell layer and inner plexiform layer, INL: Inner nuclear layer, OPL: Outer plexiform layer, ONL: Outer nuclear layer, IS: Inner photoreceptor segments, OS: Outer photoreceptor segments, RPE: Retinal pigment epithelium

TABLE II

COMPARISON OF DIFFERENT METHODS IN HISTOPATHOLOGICAL IMAGE CLASSIFICATION ON TWO DIFFERENT UNSEEN TEST DATASETS

Method	Unseen Test Dataset A					Unseen Test Dataset B				
	Accuracy	Sensitivity	Specificity	AUC	F1	Accuracy	Sensitivity	Specificity	AUC	F1
W/o Adaptation	63.15±2.21	64.65±1.73	61.62±3.17	80.72±2.24	60.13±1.78	84.37±2.32	72.30±2.15	71.32±2.63	81.98±1.33	64.04±1.56
PTBN [18]	86.17±0.44	85.20±0.30	85.08±0.93	90.68±0.90	86.58±0.15	85.36±1.54	85.79±1.01	85.03±0.76	89.93±0.65	86.18±0.54
UDA-SwAV [40]	91.40±2.00	92.39±1.67	89.54±1.84	96.24±1.61	90.96±1.23	88.64±1.25	90.29±1.72	88.91±1.87	95.24±2.31	90.67±1.46
TTT [16]	87.74±0.60	88.48±0.53	86.90±0.68	93.44±0.61	87.79±0.14	86.33±0.25	87.61±0.62	86.23±0.31	92.31±0.29	86.85±0.81
DLTTA (<i>Ours</i>)	89.03±0.95	89.51±0.64	88.24±0.75	94.67±0.69	88.19±0.42	87.69±0.25	88.52±0.34	87.74±0.39	93.65±0.46	87.03±0.26
Tent [17]	90.17±0.50	91.40±1.85	90.13±0.51	95.91±0.74	91.00±0.29	87.90±0.40	89.50±0.27	88.86±0.76	94.27±0.85	89.75±0.38
DLTTA (<i>Ours</i>)	91.49±0.07	92.68±0.12	89.66±0.21	96.36±0.08	91.28±0.05	88.93±0.19	90.59±0.14	88.62±0.28	95.15±0.06	90.58±0.13
ATTA [21]	89.69±1.24	90.89±1.99	90.56±1.73	94.23±1.22	89.17±1.17	87.54±1.23	89.03±1.67	87.42±1.69	92.58±1.86	88.52±1.74
DLTTA (<i>Ours</i>)	91.30±1.51	91.31±1.10	91.92±1.53	95.20±1.54	90.83±1.58	88.41±1.03	91.82±1.25	88.98±1.83	94.28±1.61	89.73±1.34

2) *Results for Histopathological Image Classification:* Table II shows the comparison results on the large histopathological image dataset for tumor and normal tissue classification. Results are reported on two different test datasets. We can see that even though the model is trained on images from three hospitals covering multiple data distributions, the model still performs poorly on the two unseen datasets with distribution shift, obtaining merely 63.15% accuracy on unseen dataset A. This shows the necessity of model adaptation towards the distribution of test data. Similar to the retinal layer segmentation, our dynamic learning rate strategy also consistently improves TTT, Tent and ATTA on the histopathological image classification task, showing the general applicability of our method in terms of test-time adaptation objectives, analysis tasks, and network backbones. By deploying to Tent, our method obtains 91.49% and 88.93% accuracy for the two test datasets respectively, which are comparable to the strong baseline of UDA-SwAV method utilizing both training and test data. The improvements are benefited from our adaptive learning rate adjustment to explicitly modulate the online adaptation process according to the estimated discrepancy for each test sample.

3) *Results for Prostate Segmentation:* Table III presents the prostate segmentation results with Dice score for MRI data from the five unseen test datasets B to F. We can see that

the model learned from the training dataset obtains varying performance across the five test datasets due to distribution shift. Similar to the observations in retinal layer segmentation and histopathological image classification, all test-time adaptation methods improve over the “W/o Adaptation” baseline. With the proposed dynamic learning rate strategy, our DLTTA consistently improves TTT, Tent and ATTA methods for all the test datasets. By deploying dynamic learning rate to ATTA, our method achieves 78.08% average Dice score, outperforming the unsupervised domain adaptation method UDAS which requires concurrent access to both training and test datasets. The results validate that our method is generally applicable to improve test-time adaptation with 3D models.

C. Ablation Analysis of Our Method

We conduct ablation studies to investigate several important questions regarding our dynamic learning rate strategy: 1) effectiveness of our learning rate adjustment, 2) how the adaptation stability is influenced by our method, 3) effect of initial learning rate, 4) influence of retrieval size D , 5) influence of image orders, 6) effect of similarity metrics.

1) *Effectiveness of Our Learning Rate Adjustment:* We first show that different test images require different learning rates to achieve better adaptation results. As shown in Fig. 3, for the first test case, Tent with the static learning rate $4e-3$ obtains

TABLE III
COMPARISON OF DIFFERENT METHODS IN PROSTATE SEGMENTATION ON MULTI-CENTER MRI DATASETS

Method	Dice Score in Unseen Test Datasets					Average
	Dataset B	Dataset C	Dataset D	Dataset E	Dataset F	
W/o Adaptation	80.66 \pm 2.73	70.90 \pm 2.41	73.85 \pm 3.83	57.02 \pm 2.82	81.84 \pm 2.49	72.85 \pm 2.85
PTBN [18]	80.82 \pm 2.29	72.65 \pm 2.20	73.95 \pm 3.21	61.33 \pm 2.61	82.07 \pm 3.02	74.16 \pm 2.66
UDAS [46]	81.23 \pm 1.68	75.19 \pm 1.92	76.75 \pm 2.03	65.83 \pm 1.42	86.58 \pm 1.94	77.11 \pm 1.79
TTT [16]	81.09 \pm 2.25	73.41 \pm 3.73	74.29 \pm 2.85	62.58 \pm 1.97	82.49 \pm 2.56	74.77 \pm 2.67
DLTTA (<i>Ours</i>)	83.14\pm2.34	73.63\pm2.11	76.41\pm2.14	65.54\pm2.84	83.42\pm2.77	76.44\pm2.44
Tent [17]	81.36 \pm 1.57	76.10 \pm 1.48	75.02 \pm 3.99	62.23 \pm 1.85	81.19 \pm 3.63	75.17 \pm 2.50
DLTTA (<i>Ours</i>)	82.47\pm2.20	77.84\pm2.81	76.97\pm3.05	66.32\pm2.92	82.90\pm3.81	77.30\pm2.95
ATTA [21]	81.19 \pm 3.60	75.94 \pm 3.02	74.49 \pm 3.04	68.06 \pm 2.28	85.09 \pm 2.17	76.95 \pm 2.82
DLTTA (<i>Ours</i>)	82.13\pm2.47	76.50\pm2.45	75.33\pm3.57	70.21\pm2.02	86.25\pm2.09	78.08\pm2.52

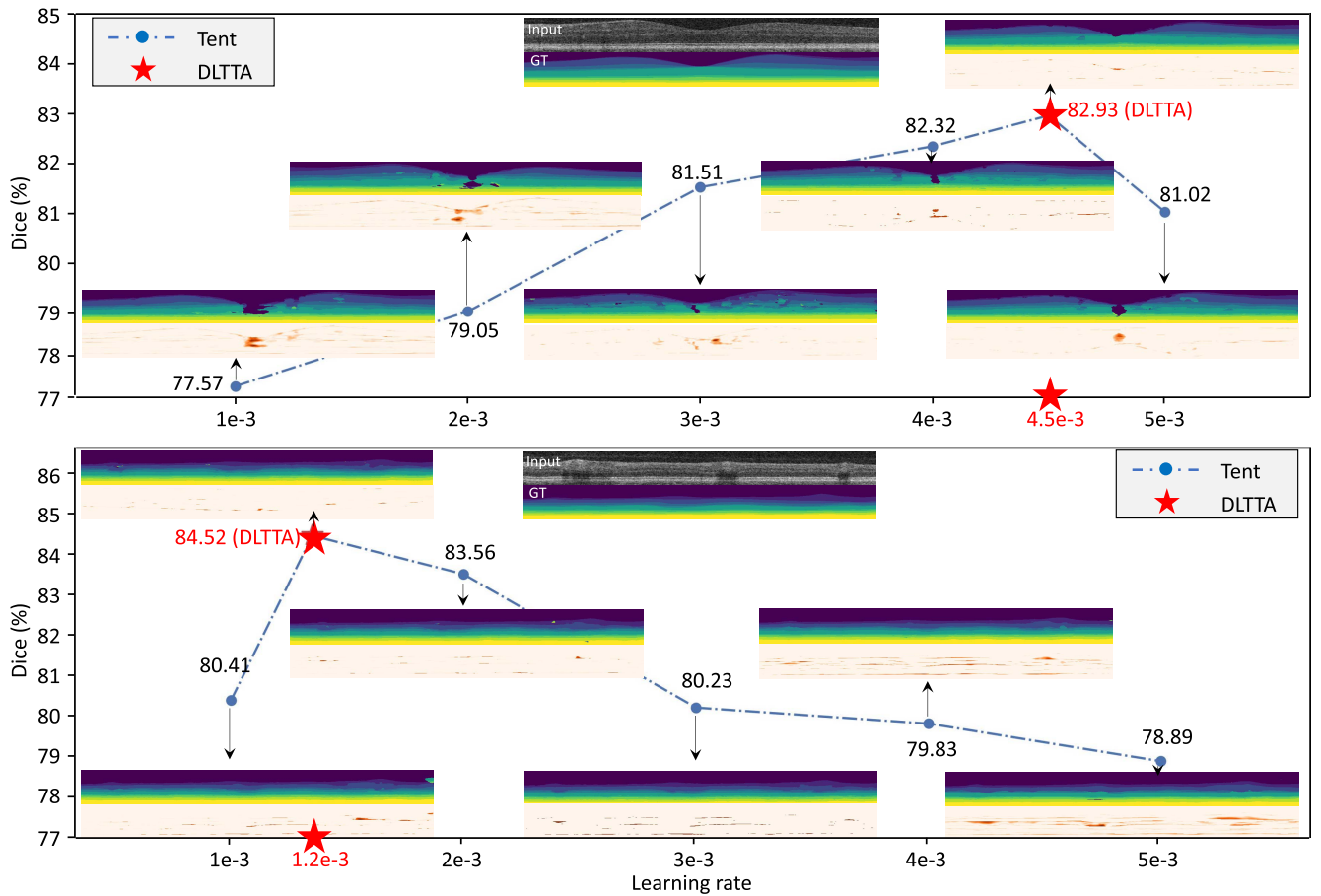


Fig. 3. Dice score of the segmentation prediction produced by Tent with different static learning rates and our DLTTA, compared with the ground truth masks (GT). Each subplot aside the Dice value consists of the predicted segmentation masks (upper half, different colors represent different retinal layers) and the error map (lower half, the absolute difference between the prediction and ground truth).

better retinal layer segmentation results than the other learning rates, while for the second test case, Tent with the learning rate $1e-3$ obtains higher Dice score. This validates our motivation that a fixed learning rate can hardly suit all the test samples. Our dynamic learning strategy adaptively sets the learning rate as $4.5e-3$ for the model update on the first test case, and $1.2e-3$ for the second one, obtaining higher segmentation results for both test samples. This shows the effectiveness of our discrepancy estimation-based learning rate adjustment.

2) Adaptation Stability: We compare the adaptation stability of Tent and our dynamic learning on the retinal layer segmentation task. Fig. 4 plots the test-time loss change with the adaptation iterations. We can see that the test-time loss curve of Tent fluctuates largely between the 25 to 40 training iterations. This may because that the fixed learning rate used in Tent cannot suit the varying adaptation demand of different test samples, leading to under- or over-adaptation for some test data. With our adaptive learning rate adjustment, the

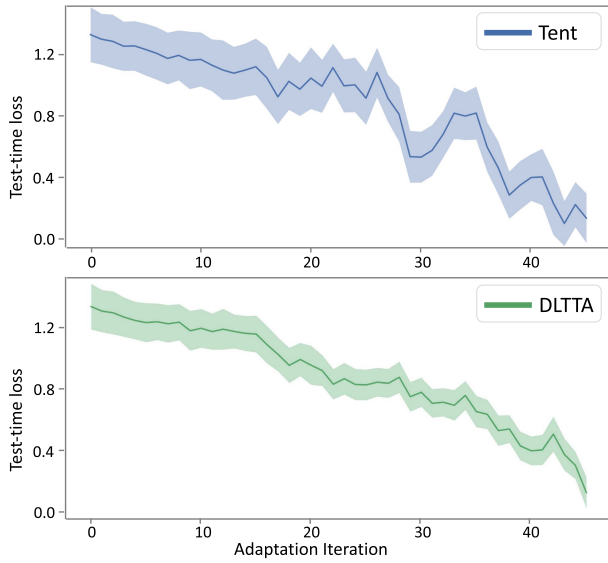


Fig. 4. Test-time loss of Tent (upper) and our DLTTA (bottom) on retinal layer segmentation task. Each plot shows the mean and standard deviation loss over all pixels of a test image, as a function of the test-time adaptation iterations. Our dynamic learning can stabilize the test-time adaptation process with smoother loss curve.

adaptation is dynamically tailored for each test sample, hence producing more stable adaptation with smoother loss curve.

3) *Effect of Initial Learning Rate*: We investigate the effect of different initial learning rates on test-time adaptation methods TTT, Tent, ATTA, and our dynamic adaptation. As shown in Table IV, with different initial learning rates from the set $\{1e-3, 2e-3, 3e-3, 4e-3, 5e-3\}$, our dynamic learning strategy consistently improves TTT, Tent and ATTA on Dice score in the retinal layer segmentation from OCT images. This indicates the general benefits of our dynamic learning rate adjustment regarding different initial learning rates. When deploying to Tent, our dynamic adaptation obtains over 82% Dice score for all different initial learning rates, which is more stable than Tent. This shows that the performance of our dynamic adaptation method is less sensitive to the choice of initial learning rate compared to previous TTA methods which adopt a fixed learning rate for all the test-time adaptation steps.

4) *The Influence of Retrieval Size*: We study how the retrieval size D for discrepancy estimation affects the performance of our method. Intuitively, less elements retrieved from the memory bank (i.e., smaller D) might be difficult to find sufficient semantically similar samples to provide prediction reference, while too more elements (i.e., larger D) could include some less relevant samples. Fig. 5 shows the change of Dice score on retinal layer segmentation when varying retrieval sizes $D \in \{6, 8, 10, 12, 14\}$. With different retrieval size, our method always obtains better performance than TTT, Tent and ATTA with the static learning strategy. The models with middle-level dictionary size ($D = 8$) perform generally better for all the three TTA objectives than the model with smaller or larger retrieval size.

5) *Influence of Image Orders*: As the test images are considered to arrive as a sequence, we study whether the image orders affect the adaptation performance. We run different

TABLE IV
COMPARISON OF TEST-TIME ADAPTATION METHODS WITH DIFFERENT INITIAL LEARNING RATES FOR RETINAL OCT IMAGE SEGMENTATION

Method	1e-3	2e-3	3e-3	4e-3	5e-3
TTT [16]	79.75	80.45	80.24	79.43	79.86
DLTTA (<i>Ours</i>)	80.54	81.47	81.09	80.41	80.67
Tent [17]	81.50	80.90	81.61	81.68	81.25
DLTTA (<i>Ours</i>)	82.63	82.36	82.48	82.46	82.27
ATTA [21]	82.50	82.77	83.21	82.42	82.21
DLTTA (<i>Ours</i>)	83.41	83.45	83.59	83.35	83.30

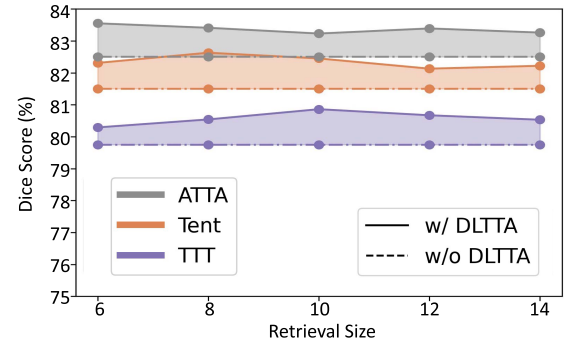


Fig. 5. The change of Dice score on retinal layer segmentation with different retrieval size of elements from the memory bank.

TABLE V
AVERAGE DICE SCORE OF DIFFERENT METHODS ON RETINAL LAYER SEGMENTATION WITH FIVE RANDOM IMAGE ORDERS

Method	Order A	Order B	Order C	Order D	Order E
TTT [16]	79.75	79.56	79.58	79.82	79.77
DLTTA (<i>Ours</i>)	80.54	80.43	80.62	80.48	80.57
Tent [17]	81.50	81.57	81.45	81.53	81.49
DLTTA (<i>Ours</i>)	82.63	82.55	82.67	82.58	82.51
ATTA [21]	82.50	82.52	82.34	82.42	82.44
DLTTA (<i>Ours</i>)	83.41	83.42	83.35	83.29	83.36

test-time adaptation methods on retinal layer segmentation with five different random orders of test images. The test dataset is shuffled before the start of the online adaptation and the image order is shared across all methods. The results presented in Table V show that the performance of our method as well as other test-time adaptation methods are not sensitive to the image orders. For our method, the largest variation is merely 0.19 in Dice.

6) *Effect of Similarity Metrics*: Since images in the memory bank are retrieved based on relative similarity order, choices of specific similarity metrics can be flexible as long as the relative order is maintained. We have compared the results of our method between using L2 distance and cosine similarity metrics on retinal layer segmentation and the dice performance difference is less than 0.1%. This shows that our method is not sensitive to the choice of similarity metric. We adopt L2 distance in this work, but other metrics such as cosine similarity metric can also be employed.

V. DISCUSSIONS

Deep neural networks are notoriously difficult to generalize to unseen domains due to data distribution shift caused by varying image acquisition conditions. This paper tackles the challenging problem of test-time adaptation, which aims to generalize the deep models to unknown data distributions by learning from the inference sample provided at test-time. Previous methods on test-time adaptation utilize a fixed learning rate for all test samples. This solution is sub-optimal since the amount that the weights should be updated could be different for the sequentially arriving test images with varying degree of distribution shift. To address this problem, in this work, we propose a dynamic learning rate strategy for test-time adaptation, aiming to dynamically adjust the step size of model update according to the estimated prediction discrepancy. Our proposed dynamic learning rate adjustment is general and can be easily applied to different test-time adaptation methods to improve performance. The general applicability and effectiveness of our method has been validated on three popular test-time objective functions, both segmentation and classification tasks, three imaging modalities, 2D and 3D models, and different network architectures.

For model learning with the gradient descent optimization algorithm, learning rate determines the step size at each iteration thus controls how much to update the weights. Learning rate is an important hyperparameter and requires careful selection and scheduling even in the standard supervised learning [33]–[36]. We consider the dynamic adjustment of learning rate becomes more significant during test-time adaptation, where a model needs to take suitable adaptation pace for the encountered new environment. In gradient descent optimization, the weights update is determined by the learning rate and the gradient of loss function. The main role of gradient of loss function is to indicate how the loss value would change given the changes of the model weights, thus determines the update direction. Although the amount of model updates can be also affected by the magnitude of gradient to some extent, it is mainly controlled by the learning rate. Therefore, loss function and learning rate play complementary role in the model learning process and both the two factors require careful design to achieve better model adaptation at test time.

Our method follows the continual online adaptation setting, where the model adaptation on test image x_t is based on the model parameters updated on the previous test sample x_{t-1} . Compared to image-specific adaptation [30], where the model is re-initialized each time with the source model parameters, continual adaptation has shown better performance in previous work [16]. This may benefit from that continual adaptation allows the model to exploit the information provided by both current image and previous images. We also implement the image-specific variants of TTA methods. Each test image is updated with 10 iterations to allow sufficient adaptation. In Table VI, we obtain similar observation to [16] that the continual adaptation outperforms the image-specific version and takes 10 times less computation. Our method can benefit more for the continual adaptation by dynamically modulating the online model update.

TABLE VI

AVERAGE DICE SCORE OF RETINAL OCT IMAGE SEGMENTATION WITH DIFFERENT TEST-TIME ADAPTATION SETTINGS

Adaptation Setting	TTT	DLTTA	Tent	DLTTA	ATTA	DLTTA
Image-specific	76.40	76.83	81.23	81.79	81.85	82.56
Image-continual	79.75	80.54	81.50	82.63	82.50	83.41

TABLE VII

AVERAGE DICE SCORE OF OUR METHOD ON RETINAL LAYER SEGMENTATION WITH DIFFERENT NUMBER OF UPDATE STEPS

Method	1	4	8
DLTTA based on TTT	80.54	80.69	80.67
DLTTA based on Tent	82.63	82.41	82.35
DLTTA based on ATTA	83.41	83.45	83.23

Existing TTA methods either take multiple adaptation steps for each test image, such as ATTA, or one gradient update, such as TTT and Tent. For the adaptation of each test image with our method, we only perform one gradient update because benefited from the proposed dynamic learning rate adjustment to explicitly module the online adaptation process, one single step can already achieve effective adaptation. We also try to update the model for multiple steps on each test image. Table VII presents the results of our methods with 1, 4, and 8 gradient updates. We can see that no consistent improvements can be obtained with multiple updates but taking more computation cost. These results demonstrate that our method can achieve effective and efficient test-time adaptation with just one gradient update, requiring less computation cost but achieving improved performance than previous methods.

One limitation of our method is that before the memory bank being constructed, it is difficult to estimate the discrepancy, thus the dynamic learning rate adjustment is not added to the test-time adaptation of the first K images. Without using dynamic learning rate for adaptation, the performance of the first K images in our method is the same as that in previous TTA methods depending on the objectives functions that the dynamic learning rate is deployed to. It would be interesting future work to explore how to obtain better discrepancy estimation for dynamic learning rate adjustment of the first several test images and to achieve faster and more effective adaptation at the beginning. A possible solution could be storing some training data feature statistics in the memory bank and adjust the learning rate by comparing training and testing feature statistics and predictions.

In general, our proposed method of dynamic learning rate strategy can be applied to many test-time learning algorithms. In this work, we mainly consider the rotation prediction, entropy minimization and autoencoder reconstruction as the objective to drive the model adaptation, while our idea to adopt dynamic step size is independent from the objective choices. For future work, we are interested in investigating the effectiveness of our method in more challenging adaptation

scenarios, such as when test images come from multiple continual changing domains.

VI. CONCLUSION

We present the first method for dynamic learning rate adjustment of test-time adaptation to effectively adapt the model towards the varying distribution shift among test data. We propose a memory bank-based discrepancy measurement by considering both the progressive model change at test time and the variations in test data distributions, and further achieve dynamic learning rate adjustment based on the estimated discrepancy. Our method is effective and generic to test-time adaptation objectives, requiring no change on the network designs, thus can be readily applied to improve different test-time adaptation methods.

REFERENCES

- [1] S. Zhao *et al.*, "A review of single-source deep unsupervised visual domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 473–493, Feb. 2022.
- [2] K. Kamnitsas *et al.*, "Unsupervised domain adaptation in brain lesion segmentation with adversarial networks," in *Proc. Int. Conf. Inf. Process. Med. Imag.* New York, NY, USA: Springer, 2017, pp. 597–609.
- [3] Y. Zhang, S. Miao, T. Mansi, and R. Liao, "Task driven generative modeling for unsupervised domain adaptation: Application to X-ray image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2018, pp. 599–607.
- [4] 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. Imag.*, vol. 39, no. 7, pp. 2494–2505, Jul. 2020.
- [5] M. Ghafoorian *et al.*, "Transfer learning for domain adaptation in MRI: Application in brain lesion segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2017, pp. 516–524.
- [6] E. Gibson *et al.*, "Inter-site variability in prostate segmentation accuracy using deep learning," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2018, pp. 506–514.
- [7] Y. Huo *et al.*, "SynSeg-Net: Synthetic segmentation without target modality ground truth," *IEEE Trans. Med. Imag.*, vol. 38, no. 4, pp. 1016–1025, Apr. 2019.
- [8] Q. Dou, C. Ouyang, C. Chen, H. Chen, and P.-A. Heng, "Unsupervised cross-modality domain adaptation of ConvNets for biomedical image segmentations with adversarial loss," in *Proc. Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 691–697.
- [9] Z. Zhang, L. Yang, and Y. Zheng, "Translating and segmenting multimodal medical volumes with cycle- and shape-consistency generative adversarial network," in *Proc. CVPR*, Jun. 2018, pp. 9242–9251.
- [10] T. Varsavsky, M. Orbes-Arteaga, C. H. Sudre, M. S. Graham, P. Nachev, and M. J. Cardoso, "Test-time unsupervised domain adaptation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2020, pp. 428–436.
- [11] L. Zhang, M. Pereañez, S. K. Piechnik, S. Neubauer, S. E. Petersen, and A. F. Frangi, "Multi-input and dataset-invariant adversarial learning (MDAL) for left and right-ventricular coverage estimation in cardiac MRI," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2018, pp. 481–489.
- [12] H. Zhao, H. Li, S. Maurer-Stroh, Y. Guo, Q. Deng, and L. Cheng, "Supervised segmentation of un-annotated retinal fundus images by synthesis," *IEEE Trans. Med. Imag.*, vol. 38, no. 1, pp. 46–56, Jan. 2019.
- [13] M. Bateson, H. Kervadec, J. Dolz, H. Lombaert, and I. B. Ayed, "Source-relaxed domain adaptation for image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2020, pp. 490–499.
- [14] C. Chen, Q. Liu, Y. Jin, Q. Dou, and P.-A. Heng, "Source-free domain adaptive fundus image segmentation with denoised pseudo-labeling," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2021, pp. 225–235.
- [15] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 7167–7176.
- [16] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *Proc. ICML*, 2020, pp. 9229–9248.
- [17] D. Wang, E. Shelhamer, S. Liu, B. A. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *Proc. ICLR*, 2021, pp. 9229–9248.
- [18] Z. Nado, S. Padhy, D. Sculley, A. D'Amour, B. Lakshminarayanan, and J. Snoek, "Evaluating prediction-time batch normalization for robustness under covariate shift," 2020, *arXiv:2006.10963*.
- [19] Y. He, A. Carass, L. Zuo, B. E. Dewey, and J. L. Prince, "Self domain adapted network," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2020, pp. 437–446.
- [20] N. Karani, E. Erdil, K. Chaitanya, and E. Konukoglu, "Test-time adaptable neural networks for robust medical image segmentation," *Med. Image Anal.*, vol. 68, Feb. 2021, Art. no. 101907.
- [21] Y. He, A. Carass, L. Zuo, B. E. Dewey, and J. L. Prince, "Autoencoder based self-supervised test-time adaptation for medical image analysis," *Med. Image Anal.*, vol. 72, Aug. 2021, Art. no. 102136.
- [22] Y. Liu, P. Kothari, B. van Delft, B. Bellot-Gurlet, T. Mordan, and A. Alahi, "TTT++: When does self-supervised test-time training fail or thrive?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 21808–21820.
- [23] Y. Iwasawa and Y. Matsuo, "Test-time classifier adjustment module for model-agnostic domain generalization," in *Proc. NeurIPS*, vol. 34, 2021, pp. 2427–2440.
- [24] P. Pandey, M. Raman, S. Varambally, and P. Ap, "Generalization on unseen domains via inference-time label-preserving target projections," in *Proc. CVPR*, Jun. 2021, pp. 12924–12933.
- [25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [26] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
- [27] X. Liu, F. Xing, C. Yang, G. E. Fakhri, and J. Woo, "Adapting off-the-shelf source segmenter for target medical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2021, pp. 549–559.
- [28] W. Zhu, Y. Huang, D. Xu, Z. Qian, W. Fan, and X. Xie, "Test-time training for deformable multi-scale image registration," in *Proc. ICRA*, May 2021, pp. 13618–13625.
- [29] X. Liu *et al.*, "Generative self-training for cross-domain unsupervised tagged-to-cine MRI synthesis," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2021, pp. 138–148.
- [30] G. Wang *et al.*, "Interactive medical image segmentation using deep learning with image-specific fine tuning," *IEEE Trans. Med. Imag.*, vol. 37, no. 7, pp. 1562–1573, Jul. 2018.
- [31] G. Valvano, A. Leo, and S. A. Tsaftaris, "Stop throwing away discriminators! Re-using adversaries for test-time training," in *Domain Adaptation and Representation Transfer, and Affordable Healthcare and AI for Resource Diverse Global Health*. New York, NY, USA: Springer, 2021, pp. 68–78.
- [32] R. Venkataramani, H. Ravishanker, and S. Anamandra, "Towards continuous domain adaptation for healthcare," 2018, *arXiv:1812.01281*.
- [33] P. Goyal *et al.*, "Accurate, large minibatch SGD: Training ImageNet in 1 hour," 2017, *arXiv:1706.02677*.
- [34] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.
- [35] D. Maclaurin, D. Duvenaud, and R. Adams, "Gradient-based hyperparameter optimization through reversible learning," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2113–2122.
- [36] A. G. Baydin, R. Cornish, D. M. Rubio, M. Schmidt, and F. Wood, "Online learning rate adaptation with hypergradient descent," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, Apr./May 2018, pp. 1–11.
- [37] S. Li, J. Zhang, W. Ma, C. H. Liu, and W. Li, "Dynamic domain adaptation for efficient inference," in *Proc. CVPR*, Jun. 2021, pp. 7832–7841.
- [38] W. Lee, S.-Y. Byun, J. Kim, M. Park, and K. Chechil, "Unsupervised model drift estimation with batch normalization statistics for dataset shift detection and model selection," 2021, *arXiv:2107.00191*.

- [39] Y. He *et al.*, "Deep learning based topology guaranteed surface and MME segmentation of multiple sclerosis subjects from retinal OCT," *Biomed. Opt. Exp.*, vol. 10, no. 10, pp. 5042–5058, 2019.
- [40] S. Sagawa *et al.*, "Extending the WILDS benchmark for unsupervised adaptation," 2021, *arXiv:2112.05090*.
- [41] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, 2017, pp. 4700–4708.
- [42] O. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2016, pp. 424–432.
- [43] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.
- [44] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. CVPR*, Jun. 2016, pp. 2414–2423.
- [45] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2223–2232.
- [46] Y.-H. Tsai, W.-C. Hung, S. Schuster, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7472–7481.
- [47] Y. He, A. Carass, S. D. Solomon, S. Saidha, P. A. Calabresi, and J. L. Prince, "Retinal layer parcellation of optical coherence tomography images: Data resource for multiple sclerosis and healthy controls," *Data Brief*, vol. 22, pp. 601–604, Feb. 2019.
- [48] P. W. Koh *et al.*, "Wilds: A benchmark of in-the-wild distribution shifts," in *Proc. ICML*, 2021, pp. 5637–5664.
- [49] P. Bándi *et al.*, "From detection of individual metastases to classification of lymph node status at the patient level: The CAMELYON17 challenge," *IEEE Trans. Med. Imag.*, vol. 38, no. 2, pp. 550–560, Feb. 2019.
- [50] G. Litjens *et al.*, "1399 H&E-stained sentinel lymph node sections of breast cancer patients: The CAMELYON dataset," *GigaScience*, vol. 7, no. 6, 2018, Art. no. giv065.
- [51] Q. Liu, Q. Dou, and P.-A. Heng, "Shape-aware meta-learning for generalizing prostate MRI segmentation to unseen domains," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.* New York, NY, USA: Springer, 2020, pp. 475–485.
- [52] B. Nicholas *et al.*, "NCI-ISBI 2013 challenge: Automated segmentation of prostate structures," *Cancer Imag. Arch.*, vol. 370, no. 6, p. 5, 2015.
- [53] G. Litjens *et al.*, "Evaluation of prostate segmentation algorithms for MRI: The PROMISE12 challenge," *Med. Image Anal.*, vol. 18, no. 2, pp. 359–373, 2014.
- [54] G. Lemaître, R. Martí, J. Freixenet, J. C. Vilanova, P. M. Walker, and F. Meriaudeau, "Computer-aided detection and diagnosis for prostate cancer based on mono and multi-parametric MRI: A review," *Comput. Biol. Med.*, vol. 60, pp. 8–31, May 2015.