

Domain Generalization Using Category Information Independent of Domain Differences

Reiji Saito^a and Kazuhiro Hotta^b

Meijo University, 1-501 Shiogamaguchi, Tempaku-ku, Nagoya 468-8502, Japan

Keywords: Domain Generalization, Segmentation, DeepCCA, SQ-VAE.


Abstract: Domain generalization is a technique aimed at enabling models to maintain high accuracy when applied to new environments or datasets (unseen domains) that differ from the datasets used in training. Generally, the accuracy of models trained on a specific dataset (source domain) often decreases significantly when evaluated on different datasets (target domain). This issue arises due to differences in domains caused by varying environmental conditions such as imaging equipment and staining methods. Therefore, we undertook two initiatives to perform segmentation that does not depend on domain differences. We propose a method that separates category information independent of domain differences from the information specific to the source domain. By using information independent of domain differences, our method enables learning the segmentation targets (e.g., blood vessels and cell nuclei). Although we extract independent information of domain differences, this cannot completely bridge the domain gap between training and test data. Therefore, we absorb the domain gap using the quantum vectors in Stochastically Quantized Variational AutoEncoder (SQ-VAE). In experiments, we evaluated our method on datasets for vascular segmentation and cell nucleus segmentation. Our methods improved the accuracy compared to conventional methods.


1 INTRODUCTION

Semantic segmentation is a technique for classifying images at the pixel level and is applied in various fields such as medical imaging (J.Wang et al., 2020; F.Milletari et al., 2016), autonomous driving (Y.Liu et al., 2020), and cellular imaging (Furukawa and Hotta, 2021; Shibuya and Hotta, 2020). Conventional methods (P.Wang et al., 2023; B.Cheng et al., 2022) are typically trained on specific datasets (source domains) and evaluated on the same datasets. However, these methods often perform poorly when evaluated on different datasets (target domains) due to domain shift. In medical segmentation, domain shift is particularly pronounced because images are captured in various hospitals and clinical settings. Domain shift occurs due to differences in imaging conditions, such as imaging devices, lighting, and staining methods. Ideally, accuracy should be maintained regardless of the dataset used for evaluation. Addressing this domain shift and effectively extracting category information that is independent of these differences is a long-standing challenge in deep learning.

One common approach to solving the domain shift problem is domain adaptation (DA). DA leverages labeled data from the source domain to adjust its distribution to match the target domain, maximizing performance on the target domain. However, this approach requires capturing and learning from target domain images, which can be time-consuming. Additionally, DA is only applicable to the specific target domain images being trained, lacking generalizability. Furthermore, in segmentation tasks, manual annotation is required, which can be a significant burden for researchers.

Domain generalization (DG) has been proposed to address the limitations of DA. DG leverages only the source domain to extract features that are not specific to it (e.g., cell nuclei and blood vessels), thereby mitigating domain shift when encountering unseen target domains. Here, we focus on developing a model that effectively generalizes across diverse medical imaging conditions, enhancing robustness and adaptability to varying environments. Research on DG (S.Choi et al., 2021; X.Pan et al., 2018) has developed methods that eliminate domain-specific style information from images and use content information for learning. Specifically, these methods involve whitening style-

^a  <https://orcid.org/0009-0003-5197-8922>

^b  <https://orcid.org/0000-0002-5675-8713>

specific features based on the correlation of feature values, thereby retaining content information and improving generalization performance. However, WildNet (S.Lee et al., 2022) noted that style information also contains essential features for semantic category prediction, and addressing this issue has been reported to improve accuracy.

To address DG without removing style information, we have employed the following two approaches. First, we proposed a method to split feature maps into two parts: domain-invariant category information and source domain-specific information. Specifically, we divide the feature maps along the channel dimension and use DeepCCA (G.Andrew et al., 2013) to decorrelate these parts. DeepCCA maximizes the correlation between two variables, but we train it to make the correlation zero to extract source domain-specific information and domain-invariant category information. We train one of the split feature maps to represent domain-invariant category information. Specifically, we train the feature vectors of the same category, based on the ground truth labels, to approach a learnable representative vector. Since the two feature maps are decorrelated, the other feature map becomes the source domain-specific information. We use the obtained domain-invariant category features for segmentation. Second, although we extracted domain-invariant category information, this alone cannot completely prevent the domain gap between the source and target domains. Therefore, we propose a method to mitigate the domain gap using quantum vectors from SQ-VAE (Y.Takida et al., 2022). SQ-VAE is a method for reconstructing high-resolution input images, capable of representing images using only quantum vectors.

Figure 1 shows the overview of DG using SQ-VAE. In Step 1, we divide N quantum vectors into K groups, where N represents the number of quantum vectors and K represents the number of categories. When an input feature is assigned to the quantum vector defined as category 0, the feature is predicted as category 0. In Step 2, we use ground truth labels to group the features into K categories and train the model to bring these groups of the same category closer together. Step 3 is inference. Since we do not have access to the target domain, a domain gap arises. However, by aligning the groups of each category, we can minimize the domain gap. Even if there is a gap in the unseen target domain, the features of the target domain are likely to be assigned to the same or similar quantum vectors as those in the source domain and thus categorized similarly. By using these methods, we can prevent accuracy degradation due to unseen target domains.

Experiments were conducted to segment blood vessels from retinal image datasets (Drive (J.Staal et al., 2004), Stare (A.Hoover et al., 2000), Chase (G.Jiaqi et al.,)). Each dataset has a different domain due to varying imaging devices. Two retinal image datasets were used for training, while the remaining one was utilized for evaluation. The proposed method achieved an average improvement of 1.36% in mIoU compared to the original U-Net when it served as the feature extractor, with a notable average increase of 2.71% in vascular regions. When we use UCTransNet as a feature extractor, the proposed method improved mIoU by 1.02% over the original UCTransNet, with a significant improvement of 2.17% in vascular regions.

Another experiment was conducted on MoNuSeg (N.Kumar et al., 2020) dataset, which exhibits diversity in nuclei across multiple organs and patients and is captured under varying staining methods at different hospitals. Therefore, DG is required to extract features that are independent of domain differences. Compared to the original U-Net (O.Ronneberger et al., 2015), our method using U-Net as a feature extractor improved mIoU by 2.53%, with an improvement of 2.73% in cell nuclei. Additionally, compared to UCTransNet (H.Wang et al., 2022), the proposed method using UCTransNet as a feature extractor also improved mIoU by 3.0%, with an improvement of 3.65% in cell nuclei.

The structure of this paper is as follows. Section 2 describes related works. Section 3 explains the details of the proposed method. Section 4 presents and discusses the experimental results. Section 5 describes conclusions and future work.

2 RELATED WORKS

2.1 Domain Generalization for Semantic Segmentation

DG only allows access to the source domain and extracts features that are not specific to it (e.g., background or cell nuclei), mitigating domain shifts for unseen target domains. Existing research on DG for semantic segmentation often focuses on methods that remove domain-specific style information. Techniques such as normalization (X.Pan et al., 2018; S.Bahmani et al., 2022), whitening (S.Choi et al., 2021), and diversification (Y.Zhao et al., 2022; D.Peng et al., 2021) have been used to achieve DG. However, WildNet argued that style information and content information are not orthogonal, and whitening style information can inadvertently remove nec-

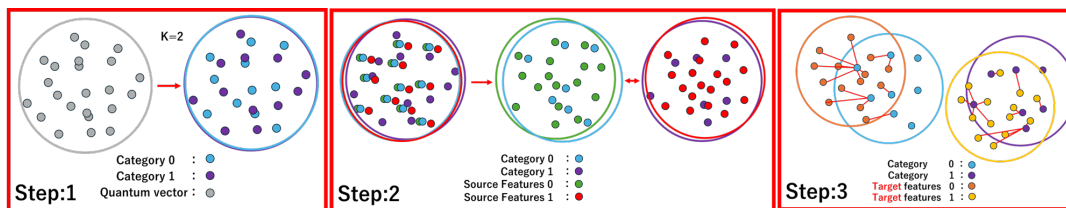


Figure 1: Overview of domain generalization using quantum vectors. This figure explains learning method for quantum vectors used to absorb domain gaps and method to handle unseen target domains during inference.

essary content information. Therefore, to build a high-precision model, it is essential to effectively acquire information that is independent of the dataset. We propose a method that splits the feature maps from input images into two parts and trains them to decorrelate from each other. One feature map is constrained to acquire category information independent of domain differences, while the other retains source domain-specific information. This approach allows effective segmentation using feature maps that contain category information independent of domain differences.

2.2 Image Generation Model

Research on generative models has been extensive, with various approaches proposed, such as Variational Autoencoder (VAE) (Kingma and Welling,), and Vector Quantized VAE (VQ-VAE) (den Oord et al., 2017). VAE maps input data to a latent space as a probability distribution and samples latent variables from this distribution to generate images. VQ-VAE possesses higher quality image generation capabilities and clustering abilities compared to VAE. However, VQ-VAE has a non-differentiability issue due to the use of the argmax function for discretization. This problem was addressed by SQ-VAE. SQ-VAE uses Gumbel-Softmax (E.Jang et al., 2016) to approximate a categorical distribution in a differentiable manner, allowing uninterrupted backpropagation. We propose a method focusing on the clustering capability of SQ-VAEs to bridge the domain gap between a source domain and an unseen target domain. Specifically, we constrain the probabilities to divide the N quantum vectors into K groups. Aligning the groups for each category can minimize the gap with the unseen target domain. This approach helps mitigate the accuracy degradation caused by the target domain.

3 PROPOSED METHOD

When conventional semantic segmentation learns from a specific dataset (source domain) and eval-

uates on an unseen dataset (target domain), accuracy decreases significantly due to domain differences such as different imaging devices and staining methods. To improve accuracy through DG, we propose two approaches. First, we separate feature maps into domain-independent category information and domain-specific information of the source domain. Using domain-independent category information for segmentation, we believe that accuracy is independent of the dataset domain used for training. Second, using only category information cannot completely bridge the domain gap between the source and target domains. Therefore, we propose a method to address the domain gap using the quantum vectors of SQ-VAE.

3.1 Category Information Independent of Domain Differences

Figure 2 illustrates the overview of our method. To extract information independent of domain differences, an input image $x \in \mathbb{R}^{3 \times H \times W}$ is processed by a feature extractor, such as U-Net or UCTransNet, which outputs the feature map $Z \in \mathbb{R}^{C \times H \times W}$. The feature extractor, such as U-Net or UCTransNet, achieves high accuracy and provides output images of the same dimensions as the input images. This is why we use U-Net or UCTransNet as the encoder. The feature maps obtained from the encoder are divided along the channel dimension into two parts: the domain-independent category information and the remaining information. The divided feature maps are denoted as $Z_1 \in \mathbb{R}^{C_1 \times H \times W}$ and $Z_2 \in \mathbb{R}^{C_2 \times H \times W}$ where $C_1 = C_2 = C/2$. To separate domain-independent category information from source domain-specific information, the model is trained to decorrelate the feature maps Z_1 and Z_2 . In this paper, we adopt DeepCCA, which can learn nonlinear relationships, allowing it to handle more complex data and achieve high precision in removing correlations.

DeepCCA maximizes the correlation between two variables in a nonlinear manner. However, since our goal is to decorrelate them, squaring the output of

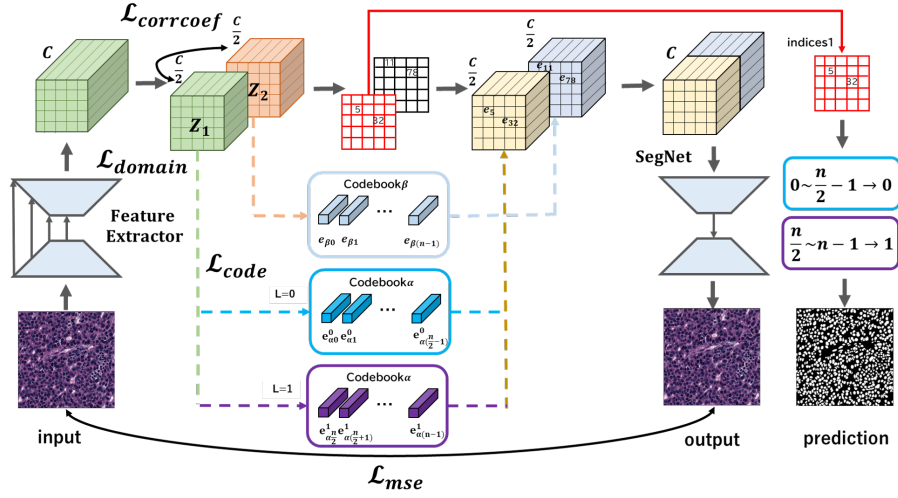


Figure 2: Overview of the proposed method. We extracted domain-independent category information to address unseen target domains. Domain-independent category information is represented using Z_1 , which is used for segmentation. Additionally, quantum vectors are used for quantization to bridge the gap between source domain and unseen target domain.

DeepCCA brings the result closer to zero.

$$\mathcal{L}_{corrcoef} = corrcoef(Z_1, Z_2)^2 \quad (1)$$

After decorrelation, feature maps are quantized using quantization vectors $e_\alpha \in \mathbb{R}^{N \times C_1}$ and $e_\beta \in \mathbb{R}^{N \times C_2}$. N is the dimension of the embedding vector space. The details are presented in Section 3.2. The reason for preparing e_α and e_β is to assign them different roles. e_α is used to provide domain-independent categorical features, while e_β supplies source domain-specific features. This approach allows each to serve distinct functions. The feature maps after quantization are denoted as Z'_1 and Z'_2 . Z'_1 and Z'_2 are also trained to be decorrelated using DeepCCA.

$$\mathcal{L}_{corrcoef} = corrcoef(Z'_1, Z'_2)^2 \quad (2)$$

By using constraints in Equations 1 and 2, the quantization vectors e_α and e_β become automatically uncorrelated. By training Z_1 and Z_2 to be uncorrelated, they can assume different roles. For example, if we assume that Z_1 and e_α represent domain-independent category features, then Z_2 and e_β represent source domain-specific features. We hypothesize that using these domain-independent features for segmentation will improve accuracy. This method involves training multiple features to be uncorrelated, but there is no guarantee that Z_1 represents domain-independent category information.

To address this, we propose a method to group features within Z_1 that belong to the same category, embedding domain-independent category information. Specifically, we introduce a set of learnable representative vectors, $t^K \in \mathbb{R}^{K \times C_1}$. Here, K denotes the number of categories. For example, for category t^0 , we

train the model to cluster features Z_1^0 in Z_1 with a target label of 0. This process is repeated for all K categories. Additionally, to ensure that these representative vectors do not capture source domain-specific information, we train the model to separate the representative vectors from each other. By doing so, we can densely embed domain-independent category information.

$$\mathcal{L}_{domain} = \sum_{i=0}^{K-1} \left\{ \|t^i - Z_1^i\|_2 - \sum_{j=0}^{K-1} \frac{\|t^i - t^j\|_2}{2} \right\} \quad (3)$$

where Z_1^0 and Z_1^1 are defined as the features of Z_1 when the target label is 0 and 1.

3.2 Segmentation Using SQ-VAE

In Section 3.1, we proposed a method that divides features into those related to domain-independent categories and those specific to the source domain. Although only domain-independent features provide DG capabilities, this does not completely prevent the domain gap. Thus, we propose a method to bridge the domain gap between the source and unseen target domains using quantum vectors from the SQ-VAE. The reason for using SQ-VAE is that it can generate various clusters by utilizing quantum vectors. Additionally, SQ-VAE is more accurate as an image generation model compared to VAE and VQ-VAE. We constrain the N quantum vectors from the SQ-VAE into K groups to address the domain gap. For instance, in the category of cell nuclei, we train the model to bring the group of quantum vectors representing cell nuclei closer to the group of features obtained from input images of cell nuclei. By aligning these groups,

we can mitigate the domain gap between the source and unseen target domains, allowing for better generalization. We feed an image $x \in \mathbb{R}^{3 \times H \times W}$ into the feature extractor, such as U-Net and UCTransNet, which outputs a feature map $Z \in \mathbb{R}^{C \times H \times W}$. As explained in Section 3.1, to separate domain-independent category features from domain-specific features, we divide the feature map into two parts: Z_1 and Z_2 .

To address the domain gap, we define the quantum vectors as $e \in \mathbb{R}^{N \times C/2}$ where $C/2$ is the channel dimension of the embedded vector e . We separate domain-independent category information from domain-specific in $e_\alpha \in \mathbb{R}^{N \times C_1}$ and $e_\beta \in \mathbb{R}^{N \times C_2}$. To quantize the feature map obtained from the feature extractor, we calculate the Mahalanobis distance between the feature map Z_1 and the quantum vector e_α .

$$\text{logit}_1 = - \left\{ \frac{(e_{\alpha j} - Z_1)^\top \Sigma_\gamma^{-1} (e_{\alpha j} - Z_1)}{2} \right\}_{j=0}^{N-1} \quad (4)$$

where Σ_γ is a learnable parameter, and j refers to one of the quantum vectors within the set of N vectors. It is denoted as $\Sigma_\gamma = \sigma_\gamma^2 I$. Then, logit_1 is expressed as

$$\text{logit}_1 = \frac{\|e_{\alpha j} - Z_1\|_2^2}{2\sigma_\gamma^2} \quad (5)$$

where $\text{logit}_1 \in \mathbb{R}^{HW \times N}$ is a matrix. In this case, $\Sigma_\gamma = \sigma_\gamma^2 I$ is learned to approach zero from the initial value. As training progresses, the probabilities of the distances between feature maps obtained by the encoder and the quantum vectors become closer to a one-hot encoding. This is similar to SQ-VAE. To probabilistically quantify the Mahalanobis distance between the obtained feature map and the quantum vectors, we use Gumbel-Softmax. We use Gumbel-Softmax because it approximates the selection of discrete quantum vectors as a continuous probability distribution and is differentiable.

$$P_1 = \text{Gumbelsoftmax} \left(-\frac{\text{logit}_1}{\tau} \right) \quad (6)$$

where τ is a learnable temperature parameter. Similarly, we use Z_2 and e_β to output P_2 . As shown in Equations 5 and 6, we calculate the Mahalanobis distance and convert it to probabilities using Gumbel-Softmax. These probabilities are then used to quantize the features obtained from the encoder.

During evaluation, we replace the feature map obtained from Equation 5 with the quantum vector that has the closest Mahalanobis distance.

$$\text{indices}_1 = \text{argmax}(\text{logit}_1) \quad (7)$$

where argmax is used along the N -dimensional direction of $\text{logit}_1 \in \mathbb{R}^{HW \times N}$.

The feature maps fed into the decoder are quantized as $Z'_1 \in \mathbb{R}^{C_1 \times H \times W}$ and $Z'_2 \in \mathbb{R}^{C_2 \times H \times W}$. Since the channel dimension is split into two, the dimensions from Z'_1 and Z'_2 are combined.

$$Z' = \text{Concat}(Z'_1, Z'_2) \quad (8)$$

According to Equation 8, it becomes $Z' \in \mathbb{R}^{C \times H \times W}$. The decoder shown in Figure 2 uses an encoder-decoder CNN (V.Badrinarayanan et al., 2017) without skips. This is because U-Net or UCTransNet features transmission mechanisms that allow the input image to flow through easily, simplifying reconstruction and hindering the learning of intermediate layers. The input Z' is passed through the encoder-decoder CNN, producing the final output x' . The reconstruction error between the input image x and the final output image x' is then calculated.

$$\mathcal{L}_{mse} = \frac{\sum_{i=1}^n \log(x_i - x'_i)^2}{2} \quad (9)$$

where n is the total number of pixels in the input image. The reason for adding log to the loss is that the gradient becomes larger as x decreases. In other words, we believe that the model should focus on finer details during reconstruction. This is consistent with the implementation of SQ-VAE.

The objective is to use features related to categories that are independent of domain differences and employ quantum vectors to prevent domain gaps for unseen target domains. Additionally, segmentation is performed by the indices of the quantum vectors. This is executed using Z_1 and $e_\alpha \in \mathbb{R}^{N \times C_1}$. It is divided into K parts to separate roles based on the number of segmentation categories K .

In this paper, we consider the case where $K=2$. First, it is divided into two parts: $e_\alpha^0 \in \mathbb{R}^{N_1 \times C_1}$, corresponding to category label 0, and $e_\alpha^1 \in \mathbb{R}^{N_2 \times C_1}$, corresponding to category label 1. Let N_1 and N_2 be such that $N_1 = N_2 = N/2$. From Equation 6, we have $P_1 \in \mathbb{R}^{H \times W \times N}$. When the category label is 0, we want the probability of selecting e_α^0 to be 1. Similarly, when the category label is 1, we want the probability of selecting e_α^1 to be 1. To achieve this, we train the model such that the sum of probabilities in the N_1 dimensional direction of P_1 is 1 when the category label is 0, and the sum of probabilities in the N_2 dimensional direction of P_1 is 1 when the category label is 1. In other words, when the category label is 0, the model is trained to minimize the distance between the feature map $Z_1^0 \in \mathbb{R}^{C_1 \times H_0 \times W_0}$ related to the category label and $e_\alpha^0 \in \mathbb{R}^{N_1 \times C_1}$ where H_0 and W_0 correspond to category label 0. When the category label is 1, the model is trained to minimize the distance between the feature map $Z_1^1 \in \mathbb{R}^{C_1 \times H_1 \times W_1}$ related to the category label and $e_\alpha^1 \in \mathbb{R}^{N_2 \times C_1}$ where H_1 and W_1 correspond to category

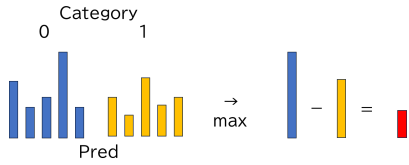


Figure 3: Weights that are learned to focus on parts where predictions become uncertain. Treat red as weight.

label 1. This learning method is as

$$\mathcal{L}_{\text{code}} = \log \left\{ w_0 \times \left(1 - \sum_{N=0}^{\frac{N}{2}-1} P_1 \right)_{L=0}^2 + w_1 \times \left(1 - \sum_{N=\frac{N}{2}}^{N-1} P_1 \right)_{L=1}^2 \right\} \quad (10)$$

where L is the label, and w_0 and w_1 are weights added to prioritize uncertain predictions. By using these weights, the model can focus on learning parts where predictions are uncertain, such as the boundaries in segmentation.

Figure 3 shows a conceptual diagram of the weights. The maximum values of the prediction probabilities from index 0 to $(N-1)/2$ and from $N/2$ to $N-1$ are obtained. These are defined as $\max(P_1^0)$ and $\max(P_1^1)$, respectively. Next, the absolute value of the difference between these maximum values is taken as

$$\text{dif} = |\max(P_1^0) - \max(P_1^1)| \quad (11)$$

The absolute differences are collected for all pixels and divided by the maximum value among all pixels to normalize them from 0 to 1. The weight of the i -th pixel is as follows:

$$w_i = \frac{\text{dif}_i}{\max(\text{dif})} \quad (12)$$

The weight w is less likely to be learned if it is closer to 1, as it indicates higher certainty. Conversely, if it is closer to 0, it signifies greater uncertainty in the prediction, and it is learned more intensively. The reason weights closer to 0 are learned more intensively than those closer to 1 is that Equation 10 includes a log, making smaller loss values more significant. Taking the derivative of $\log(x)$ results in $1/x$, meaning that as x becomes smaller, the gradient becomes larger, thereby making smaller loss more significant.

Finally, the index of the quantization vector closest to the feature vector is obtained, which is related to the category and independent of the domain differences. If this index is between 0 and $(N/2) - 1$, category label 0 is assigned. If the index is between $N/2$ and $N - 1$, category label 1 is assigned. This category label is used as the final segmentation prediction. The learning method using this quantized vector is shown in Figure 1, with Step 2 particularly pertaining to that part. When the category label is 0, the

model is trained so that the sum in the N_1 dimensional direction of P_1 equals 1. When the category label is 1, the model is trained so that the sum in the N_2 dimensional direction of P_1 equals 1. This approach divides the features into two groups using the labeled data, similar to dividing the quantization vectors into two groups. The model is then trained to bring the groups closer together. As a result, even in the presence of a domain gap in the unseen target domain, it can be mitigated, as quantization assigns similar quantum vectors to clusters (groups of the same category).

4 EXPERIMENTS

4.1 Implementation Details

Our method is evaluated on blood vessel segmentation from three types of fundus images: Drive, Stare, and Chase. The Drive, Stare, and Chase datasets each contain a total of 20, 20, and 28 images, respectively, along with annotations for segmenting the images into classes: background and blood vessels. The images in the Drive dataset were captured using a Canon CR5 non-mydratric 3CCD camera. The images in the Stare dataset were obtained using a Top Con TRV-50 retinal camera. The images in the Chase dataset were captured with a Nidek NM-200-D fundus camera. As these images were taken with different cameras, they can be considered to belong to different domains.

Of the three datasets, two are used for training, while the remaining dataset is used for evaluation. By rotating this arrangement, the DG performance is assessed. Each dataset is divided into five parts to perform 5-fold cross-validation, with four parts used for training (e.g., 4/5 of Drive and 4/5 of Chase) and one part used for validation (e.g., 1/5 of Drive and 1/5 of Chase). Subsequently, the two datasets used for training and validation are combined to ensure there is no data imbalance. The validation data from the remaining dataset, which was not used for training, is used for evaluation (e.g., 1/5 of Stare). However, since the Chase dataset contains more images, to avoid bias in training or evaluation, the number of images in the Chase dataset is randomly reduced to 20 for the experiments.

Additionally, we conduct experiments on the MoNuSeg dataset, which contains tissue images of tumors from various organs diagnosed in several patients across multiple hospitals. Due to the diverse appearance of nuclei across different organs and patients, as well as the variety of staining methods used by various hospitals, it is important to extract domain-agnostic information from this dataset. The

MoNuSeg dataset consists of 30 images for training and 14 images for evaluation. Among the training data, 24 images are allocated for training, and 6 images are reserved for validation. The test data is used as is for evaluation and includes lung and brain cells that are not present in the training data, rendering them unseen data.

For all experiments, the seed value is changed four times to calculate average accuracy. We resize all images to 256×256 pixels as preprocessing. The learning rate is set to 1×10^{-3} , the batch size is 2, the optimizer is Adam, and the number of epochs is 200. We used an Nvidia RTX A6000 GPU. The number of quantum vectors is set to 512. The evaluation metric is intersection over union (IoU), and we evaluate using the IoU for each class and the mean IoU (mIoU) across all classes. We compared the proposed method with U-Net and UCTransNet. The rationale is that the proposed method uses U-Net or UCTransNet as an encoder and makes predictions using quantum vectors based on its output. In other words, the same feature extractor is used up to the point of segmentation prediction.

4.2 Domain Generalization on Chase, Stare, and Drive Datasets

The results of DG on the Chase, Stare, and Drive datasets are shown in Table 1. The method with the highest accuracy is shown in orange, while the second-highest accuracy is in blue. When the Drive and Stare datasets were used for training and the Chase dataset for evaluation, the proposed method (U-Net+ours) using U-Net as a feature extractor exhibited a 1.80% improvement in mIoU compared to the original U-Net, with a specific improvement of 3.69% in the blood vessel area. Additionally, the proposed method (UCTransNet+ours) using UCTransNet as a feature extractor demonstrated a 2.41% improvement in mIoU compared to the original UCTransNet, with a specific improvement of 5.01% in the blood vessel area. When the Drive and Chase datasets were used for training and the Stare dataset for evaluation, the proposed method (U-Net+ours) using U-Net as a feature extractor showed a 1.20% improvement in mIoU compared to the original U-Net, with a specific improvement of 2.47% in the blood vessel area. Additionally, the proposed method (UCTransNet+ours) using UCTransNet as a feature extractor noted a 0.32% improvement in mIoU compared to the original UCTransNet, with a specific improvement of 0.76% in the blood vessel area. When the Stare and Chase datasets were used for training and the Drive dataset for evaluation, the proposed

Table 1: IoU and standard deviation Chase, Stare, and Drive datasets. **orange** indicates the highest accuracy, and **blue** indicates the second-highest accuracy.

datasets	methods	background	blood vessels	mIoU
Chase	U-Net	95.33(± 0.35)	43.56(± 4.14)	69.44(± 2.15)
	U-Net + ours	95.24(± 0.48)	47.25(± 1.82)	71.24(± 1.13)
	UCTransNet	95.27(± 0.37)	45.92(± 3.84)	70.59(± 1.97)
	UCTransNet + ours	95.06(± 0.39)	50.93(± 1.65)	73.0(± 0.92)
Stare	U-Net	95.81(± 0.86)	56.70(± 6.79)	76.26(± 3.73)
	U-Net + ours	95.75(± 0.71)	59.17(± 4.16)	77.46(± 2.36)
	UCTransNet	95.80(± 0.85)	56.71(± 5.66)	76.25(± 3.20)
	UCTransNet + ours	95.67(± 0.71)	57.47(± 4.17)	76.57(± 2.36)
Drive	U-Net	95.56(± 0.53)	57.86(± 1.85)	76.71(± 1.17)
	U-Net + ours	95.75(± 0.40)	59.84(± 3.06)	77.80(± 1.57)
	UCTransNet	95.62(± 0.49)	58.35(± 2.10)	76.99(± 1.28)
	UCTransNet + ours	95.58(± 0.46)	59.10(± 1.58)	77.34(± 1.01)

method (U-Net+ours) using U-Net as a feature extractor showed a 1.09% improvement in mIoU compared to the original U-Net, with a specific improvement of 1.98% in the blood vessel area. Additionally, the proposed method (UCTransNet+ours), using UCTransNet as a feature extractor showed a 0.35% improvement in mIoU compared to the original UCTransNet, with a specific improvement of 0.75% in the blood vessel area. These improvements indicate that DG is effectively achieved.

Additionally, segmentation results are shown in Figure 4. The top three rows display the results on the Chase, Stare, and Drive datasets. The areas highlighted in red boxes show significant improvements. For the Chase dataset, vascular regions in the red box of the input image appear slightly darker, which the original U-Net and UCTransNet predict as background. In contrast, our methods (U-Net+ours and UCTransNet+ours) extract category information independently of the domain, preventing domain gaps, can densely extract blood vessel category information, predicting them correctly. For the Stare dataset, focusing on the red box areas, the original U-Net and UCTransNet make predictions indicating disconnected blood vessels. However, the proposed methods (U-Net+ours and UCTransNet+ours) predict connected blood vessels, effectively extracting category information independently of the domain. For the Drive dataset, in the red box areas, the original U-Net and UCTransNet predict the blood vessels as thin or disconnected. In contrast, the proposed methods predict thicker blood vessels and connect previously disconnected vessels, successfully extracting domain-independent information.

4.3 Domain Generalization on MoNuSeg

The results of DG on the MoNuSeg dataset are shown in Table 2. The method with the highest accuracy is highlighted in orange, while the method with the second-highest accuracy is shown in blue. Compar-

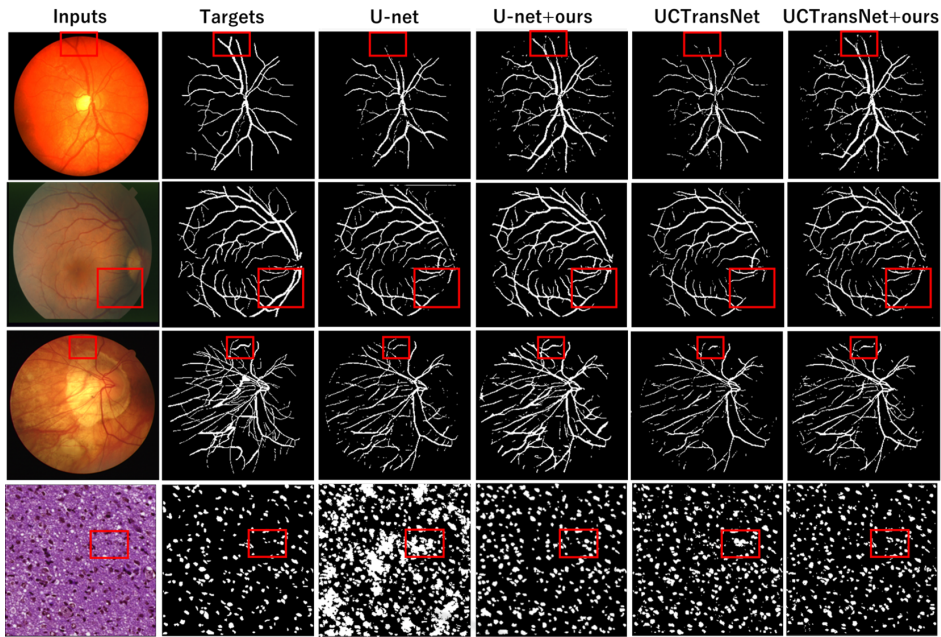


Figure 4: Segmentation results on Chase, Stare, MoNuSeg, and Drive datasets. From left to right, the images show input images, ground truth, results by original U-Net, our method (U-Net+ours), original UCTransNet, and our method (UCTransNet+ours).

Table 2: IoU and standard deviation on MoNuSeg dataset. **orange** method achieved the highest accuracy, while **blue** method attained the second-highest accuracy.

methods	background	cell nucleus	mIoU
U-Net	87.36(± 0.96)	61.55(± 1.17)	74.46(± 1.01)
U-Net + ours	89.71 (± 1.24)	64.28 (± 1.10)	76.99 (± 1.17)
UCTransNet	87.79(± 0.07)	60.93(± 0.51)	74.36(± 0.23)
UCTransNet + ours	90.15 (± 0.95)	64.58 (± 0.67)	77.36 (± 0.80)

Comparison results between conventional methods and the proposed methods (methods + ours) are included. As a result, the proposed method (U-Net+ours) achieved a 2.53% improvement in mIoU compared to the original U-Net, with a specific improvement of 2.73% in the cell nucleus area. Additionally, the proposed method (UCTransNet+ours) achieved a 3.0% improvement in mIoU compared to the original UCTransNet, with a specific improvement of 3.65% in the cell nucleus area. These results demonstrate strong generalization performance to unseen target domains. The improved accuracy in cell nuclei, in particular, suggests that the method can effectively handle the diversity in the appearance of cell nuclei and staining methods across different domains, successfully extracting cell nucleus-specific features.

Segmentation results from conventional methods and our methods are displayed in the bottom row of Figure 4. The areas highlighted in red boxes indicate where significant improvements were observed. The original U-Net struggled with DG, often predicting background as cell nuclei. In contrast, the proposed

method (U-Net+ours) successfully extracted category information independent of the domain, closing domain gaps and enabling accurate predictions. Similarly, the original UCTransNet over-predicted cell nuclei in the areas highlighted in red. However, the proposed method (UCTransNet+ours) effectively extracted information on various cell nuclei, leading to accurate predictions.

5 CONCLUSION

We proposed a method that generalizes well to datasets with differences in imaging equipment and staining methods (target domain) compared to the dataset (source domain) on which the model was trained. The proposed method showed significant improvements on cell image datasets with various staining methods and fundus images captured by different imaging devices. These results demonstrate the generalization performance of our method to unseen target domains. In the future, we would like to evaluate our method on a multi-class segmentation problem.

ACKNOWLEDGMENTS

This paper is partially supported by the Strategic Innovation Creation Program.

REFERENCES

- A.Hoover et al. (2000). Locating blood vessels in retinal images by piece-wise threshold probing of a matched filter response. *IEEE Transactions on Medical Imaging*, 19(3):203–210.
- B.Cheng et al. (2022). Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299.
- den Oord, A. et al. (2017). Neural discrete representation learning. In I.Guyon et al., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- D.Peng et al. (2021). Global and local texture randomization for synthetic-to-real semantic segmentation. *IEEE Transactions on Image Processing*, 30:6594–6608.
- E.Jang et al. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.
- F.Milletari et al. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. Ieee.
- Furukawa, R. and Hotta, K. (2021). Localized feature aggregation module for semantic segmentation. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1745–1750. IEEE.
- G.Andrew et al. (2013). Deep canonical correlation analysis. In Dasgupta, S. and McAllester, D., editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1247–1255, Atlanta, Georgia, USA. PMLR.
- G.Jiaqi et al. Chase: A large-scale and pragmatic chinese dataset for cross-database context-dependent text-to-sql.
- H.Wang et al. (2022). Uctransnet: rethinking the skip connections in u-net from a channel-wise perspective with transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 2441–2449.
- J.Staal et al. (2004). Ridge-based vessel segmentation in color images of the retina. *IEEE transactions on medical imaging*, 23(4):501–509.
- J.Wang et al. (2020). Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364.
- Kingma, D. P. and Welling, M. Auto-encoding variational bayes.
- N.Kumar et al. (2020). A multi-organ nucleus segmentation challenge. *IEEE Transactions on Medical Imaging*, 39(5):1380–1391.
- O.Ronneberger et al. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*, pages 234–241. Springer.
- P.Wang et al. (2023). One-peace: Exploring one general representation model toward unlimited modalities. *arXiv preprint arXiv:2305.11172*.
- S.Bahmani et al. (2022). Semantic self-adaptation: Enhancing generalization with a single sample. *arXiv preprint arXiv:2208.05788*.
- S.Choi et al. (2021). Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11580–11590.
- Shibuya, E. and Hotta, K. (2020). Feedback u-net for cell image segmentation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition workshops*, pages 974–975.
- S.Lee et al. (2022). Wildnet: Learning domain generalized semantic segmentation from the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9936–9946.
- V.Badrinarayanan et al. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495.
- X.Pan et al. (2018). Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the european conference on computer vision (ECCV)*, pages 464–479.
- Y.Liu et al. (2020). Efficient semantic video segmentation with per-frame inference. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 352–368. Springer.
- Y.Takida et al. (2022). SQ-VAE: Variational Bayes on discrete representation with self-annealed stochastic quantization. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S., editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 20987–21012. PMLR.
- Y.Zhao et al. (2022). Style-hallucinated dual consistency learning for domain generalized semantic segmentation. In *European conference on computer vision*, pages 535–552. Springer.