1. APPENDIX: SUPPLEMENTARY MATERIAL AND ANALYSES

To thoroughly explore and validate the efficacy and adaptability of the proposed framework, we conducted further supplementary analysis on the details that were not specified in the above paper.

1.1. Introduction and analysis of loss functions and network structure in models

In Section 2, we give a general introduction to the network structure. The loss functions and network structure in our method are described and analyzed next.

1.1.1. Registration Network

The registration network, referred to as R, processes a pair of images, denoted as (x, y), and generates a deformation field, symbolized as $\phi = R(x, y)$. This field is used to align a warped version of image x, referred to as $x(\phi)$, with image y. In a two-dimensional framework, the deformation field is composed of a matrix containing 2D vectors, each representing the direction in which every pixel in the source image x should move. To ensure that these deformation fields are smooth and to minimize the risk of excessively distorting the deformed image $x(\phi)$, an L_2-norm of the gradients of the deformation field is employed as a regularization term [1], labeled as L_{Smooth}. In a formal sense, the loss at a specific pixel, denoted as $v = (i, j)$, is calculated by:

$$L_{Smooth}(\phi, v) = \sum_{u \in N(v)} \|\phi(u) - \phi(v)\|_2 \tag{1}$$

where $N(v)$ denotes a set of neighbor pixels of the v.

1.1.2. Translation Network

The translation network, named T, is designed to process images from a source domain $\mathcal{X}$ and produce translated images that resemble those in a target domain $\mathcal{Y}$. This network T is divided into two main parts: an encoder (T_{enc}) and a decoder (T_{dec}). The encoder T_{enc} is responsible for extracting features related to the shape of the images, while the decoder T_{dec} is tasked with carrying out shape-preserving modality translation using these features. When an input image x is given, T_{enc} and T_{dec} work together to create the output y', which is defined as $y' = T(x) = T_{dec}(T_{enc}(x))$.

If the output from network T successfully preserves the shape, meaning that it does not alter the anatomical structure, then the task of aligning the images is handled exclusively by the registration network R. To ensure the consistency of shapes in the translation process, a specific loss function, known as decoupled contrastive learning (DCL) loss, is used. This loss function aims to eliminate the negative-positive-coupling (NPC) effect, thereby reinforcing shape consistency.

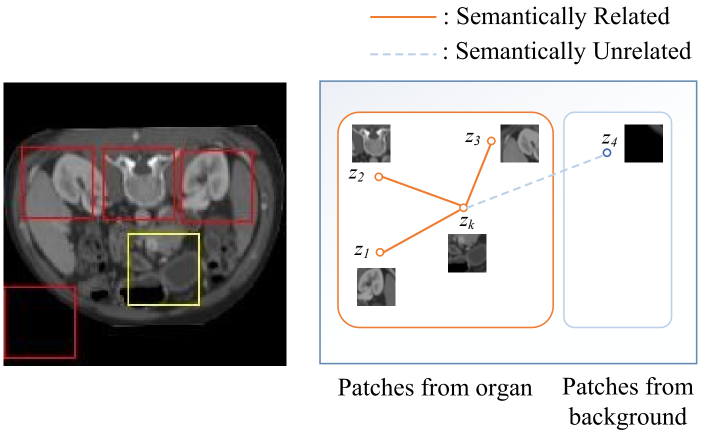

Fig. 1. Heterogeneous semantic relation between patches. The yellow patch functions as the query, while the red patches represent negatives. The patches derived from the organ exhibit a semantic relationship, whereas the background patches lack such a connection.

Additionally, a pixel loss is applied to facilitate the transfer of appearance from the source modality to the target modality.

PatchDCL loss. Due to the presence of heterogeneous semantic relationships between image patches. Since negatives are randomly sampled, it is possible for semantically unrelated image patches to be included as negatives. Despite using attention-based positive and negative sample selection, the challenge of incorporating semantically unrelated image patches still persists. For instance, as illustrated in Figure 1, z_4 represent unrelated easy negatives, which contribute to the NPC effect.

Specifically, for a positive pair (z_k, w_k), the DCL loss is defined as:

$$L_{DCL} = -\log \frac{\exp(z_k^\top w_k/\tau)}{\sum_{j \neq k} \exp(z_j^\top w_k/\tau)} \tag{2}$$

which removes the positive pair term in the denominator from the InfoNCE loss:

$$L_{InfoNCE} = -\log \frac{\exp(z_k^\top w_k/\tau)}{\exp(z_k^\top w_k/\tau) + \sum_{j \neq k} \exp(z_j^\top w_k/\tau)} \tag{3}$$

As discussed in [2], the gradient from the loss function $\nabla_{w_k} L$ for the contrastive loss L in (2) and (3) is given as:

$$\nabla_{w_k} L = -\frac{\alpha}{\tau} \left\{ z_k - \sum_{j \neq k} \frac{\exp(z_j^\top w_k/\tau)}{\sum_{m \neq k} \exp(z_m^\top w_k/\tau)} z_j \right\} \tag{4}$$

where

$$\alpha := \begin{cases} q_{NPC} & \text{if } L = L_{InfoNCE} \\ 1 & \text{if } L = L_{DCL} \end{cases} \tag{5}$$

and

$$q_{NPC} \simeq 1 - \frac{\exp(z_k^\top w_k/\tau)}{\exp(z_k^\top w_k/\tau) + \sum_{j \neq k} \exp(z_j^\top w_k/\tau)} \tag{6}$$

If z_j has no semantic relation with z_k, the value of $\exp(z_j^\top z_k/\tau)$ will be small. As a result, the denominator in equation (6) will be dominated by $\exp(z_k^\top w_k/\tau)$, leading to a decrease in the values of q_{NPC} and α according to equation (5). This, in turn, causes the gradient for InfoNCE loss to diminish. However, with DCL loss, the gradients are independent of q_{NPC} as α = 1, thus preventing the gradient from vanishing due to easy negatives. To mitigate the NPC effect, our method employs the DCL loss. In our translation network, the encoder part, T_{enc}, is enhanced with an additional two-layer multi-layer perceptron (MLP). This MLP is utilized for transforming image patches into embedded vectors. Specifically, it embeds the query, a positive sample, and N negative samples into K-dimensional vectors. These vectors are represented as z for the query, $z^+ \in R^K$ for the positive sample, and $z^- \in R^{N\times K}$ for the N negative samples, with $z_n^- \in R^K$ denoting the n-th negative patch. The process of similarity measurement is then reframed as an $(N+1)$-way classification problem. In this setup, the similarities between the query and other samples (both positive and negative) are expressed in the form of logits. A cross-entropy loss for multi-class classification is computed, which represents the probability of the positive sample being chosen over the N negative samples. The formula is given by:

$$l(z, z^+, z^-) = -\log \frac{\exp(z \cdot z^+/\tau)}{\sum_{n=1}^{N} \exp(z \cdot z^-/\tau)} \qquad (7)$$

where τ is the temperature parameter set to 0.07 in our experiments.

The encoder part, T_{enc}, of an input image processing system produces multi-layered hidden features to create a feature stack. Each spatial location within a layer of this stack corresponds to a specific patch of the input image. As you move to deeper layers, these patches represent larger areas of the image. The number of layers extracted from the feature stack is denoted as L. For each selected layer, a two-layer Multi-Layer Perceptron (MLP), named H_l, is employed to transform the encoder features from that specific layer (l-th layer) into embedded representations, symbolized as $\{z_l\}_L = \left\{H_l(T_{enc}^l(x))\right\}_L$. Here, T_{enc} indicates the features from the l-th selected layer and $l \in \{1, 2, ..., L\}$ .Let $s \in \{1, ..., S_l\}$,where S_l represents the number of spatial locations in each T_{enc}. For each spatial location s, the corresponding embedded code is referred to as $z_l^s \in R^K$, and the features at any other locations are denoted by $z_l^{S/s} \in R^{(S_l-1)\times K}$. Similarly, the corresponding embedded representations of output image y' are $\left\{z_l^{'}\right\}_L = \left\{H_l(T_{enc}^l(G(x)))\right\}_L$.

Incorporating features from various levels of the encoder, patchwise noise contrastive estimation can be implemented on different scales. The Multilayer PatchDCL loss is defined as follows:

$$L_{PatchDCL}(T, H, X) = E_x \sum_{l=1}^{L} \sum_{s=1}^{S_l} l(z_l^{'s}, z_l^s, z_l^{S/s}) \qquad (8)$$

In addition, $L_{PatchDCL}$ is also used on images from target domain $\mathcal{Y}$, which acts as reconstruction loss. Through the loss $L_{PatchDCL}(T, H, Y)$, network T outputs the reconstruction $\hat{y} = T(y)$.

Pixel Loss. Decouple contrastive loss is successful in maintaining the shape of the input image x. Nevertheless, it's also necessary to enhance the appearance similarity between a translated image y' and the target domain $\mathcal{Y}$ using a pixel loss. The pixel loss is defined using the $L1$-norm as follows:

$$L_{appearance}(T, R) = \|y'(\phi) - y\|_1 \qquad (9)$$

where $y'(\phi)$ indicates the warped image of y.

$L_{appearance}$ explicitly imposes penalties on the absolute intensity variations between $y'(\phi)$ and y. The integration of $L_{appearance}$ and $L_{PatchDCL}$ results in a translation process that is free from discriminators and preserves the shape. It is important to note that through the minimization of these two losses, the registration network R is trained collaboratively to predict a deformation field ϕ, which aligns y' with y.

Local-DCL Loss. In order to facilitate the learning of local (patch-level) alignment by R, we introduce a modified version of the PatchDCL loss. Similarly, we designate a patch from the transformed source image $x(\phi)$ as a "query," while "positives" and "negatives" refer to corresponding and non-corresponding patch(es) within the target image y, respectively. Both the transformed source image and the target image are represented as embedded vectors $\{q_l\}_L = \left\{H_l(T_{enc}^l(x(\phi)))\right\}_L$ and $\{p_l\}_L = \left\{H_l(T_{enc}^l(y))\right\}_L$, respectively. The computation of patchwise noise contrastive estimation is performed between the embedded vectors $\{q_l\}_L$ and $\{p_l\}_L$, which distinguishes it from the original PatchDCL. To maintain clarity, we refer to this modified loss as $L_{local-DCL}$:

$$L_{local-DCL}(R) = E_{x,y} \sum_{l=1}^{L} \sum_{s=1}^{S_l} l(q_l^s, p_l^s, p_l^{S/s}) \qquad (10)$$

Global Loss. The above describes the application of $L_{local-DCL}$ on cross-modality image patches, which aids the registration network R in learning local alignment. It's noted that the images generated by the translation network lack texture information, which is advantageous for extracting global information. Building on this insight, a global alignment loss is further proposed as L_{global}:

$$L_{global}(T, R) = \|y'(\phi) - \hat{y}\|_1 \qquad (11)$$

By reducing L_{global}, the style of the generated y' and $\hat{y}$ becomes more alike. At the same time, the registration network R is trained to learn a deformation field ϕ, which effectively aligns y' with $\hat{y}$.

2. REFERENCES

[1] Andrew Hoopes, Malte Hoffmann, Bruce Fischl, John Guttag, and Adrian V Dalca, "Hypermorph: Amortized

hyperparameter learning for image registration," in *IPMI*. Springer, 2021, pp. 3–17.

[2] Chun-Hsiao Yeh, Cheng-Yao Hong, Yen-Chi Hsu, Tyng-Luh Liu, Yubei Chen, and Yann LeCun, "Decoupled contrastive learning," in *ECCV*. Springer, 2022, pp. 668–684.