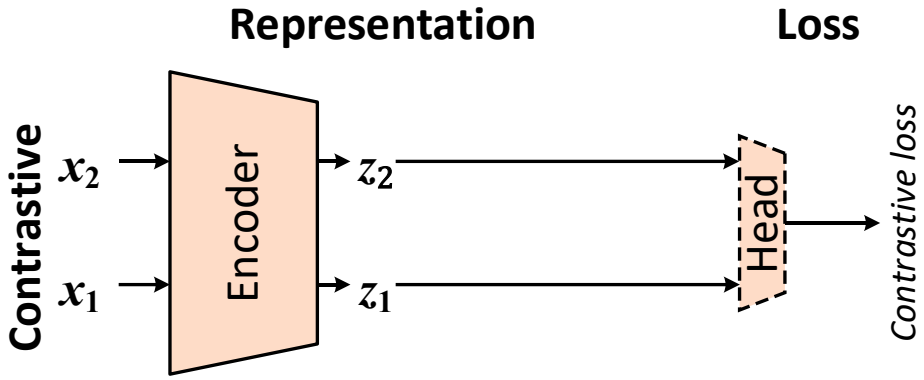
Vector Contrastive Learning

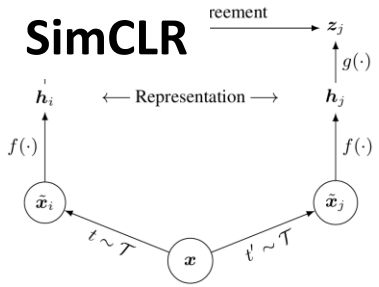
For Pixel-Wise Pretraining In Medical Vision

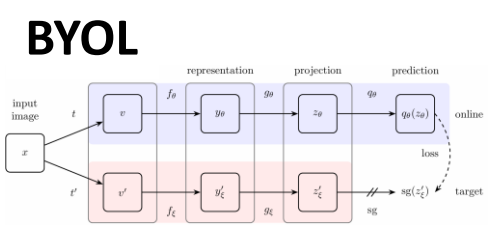
Yuting (Bruce) He 何宇霆

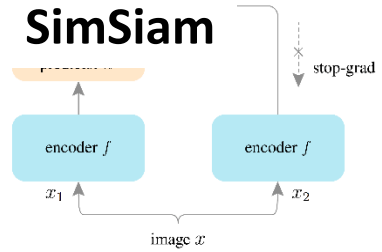
Background – What is contrastive learning?

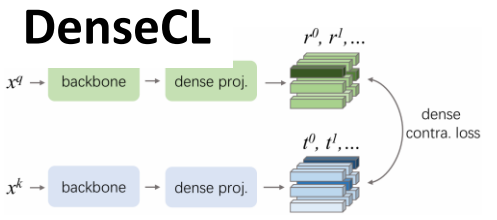


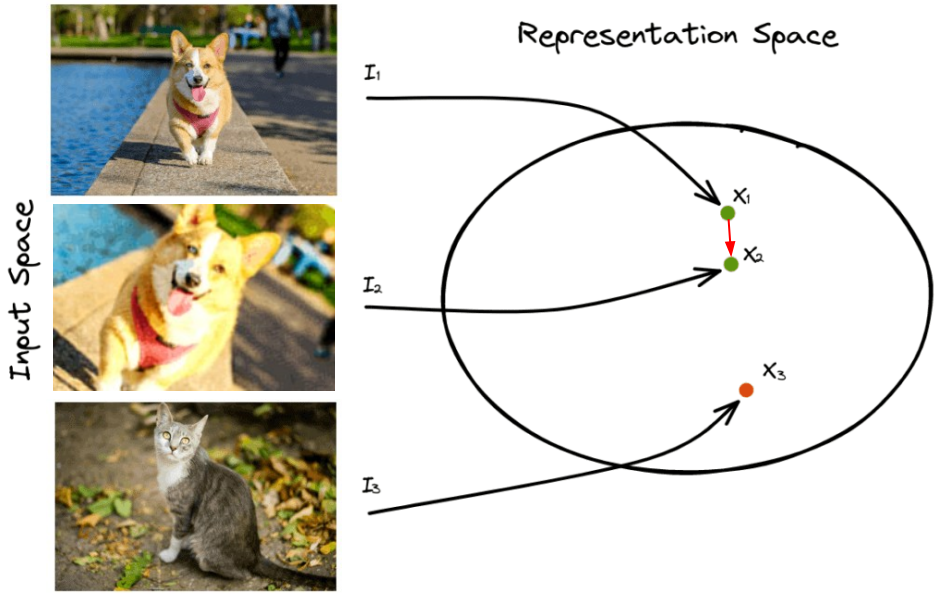
- **Contrastive:** train an encoder to encode input x into an explicit vector z to measure similarity (e.g., mutual information maximization, instance discrimination)

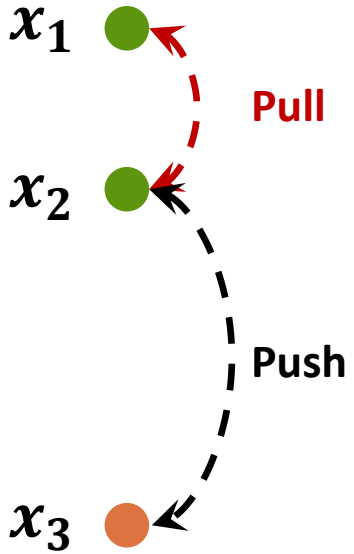








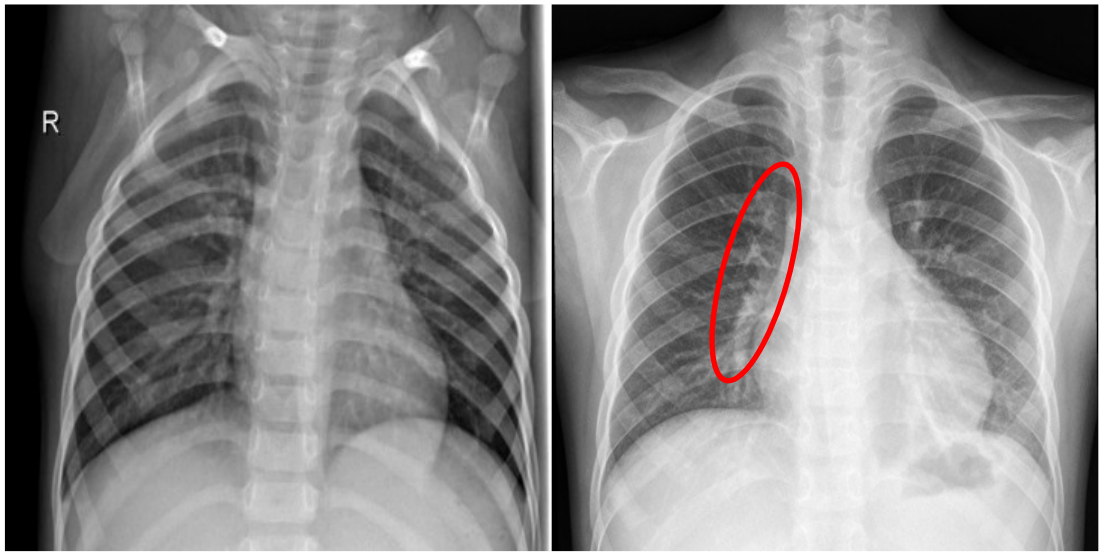




Background – Why is pixel-wise pre-training in medical vision needed?

Train *Car*

i. Natural images are **different in global** enable **global-based discrimination**



Normal *Pneumonia*

ii. Medical images are **similar in global** require **detail-based discrimination**

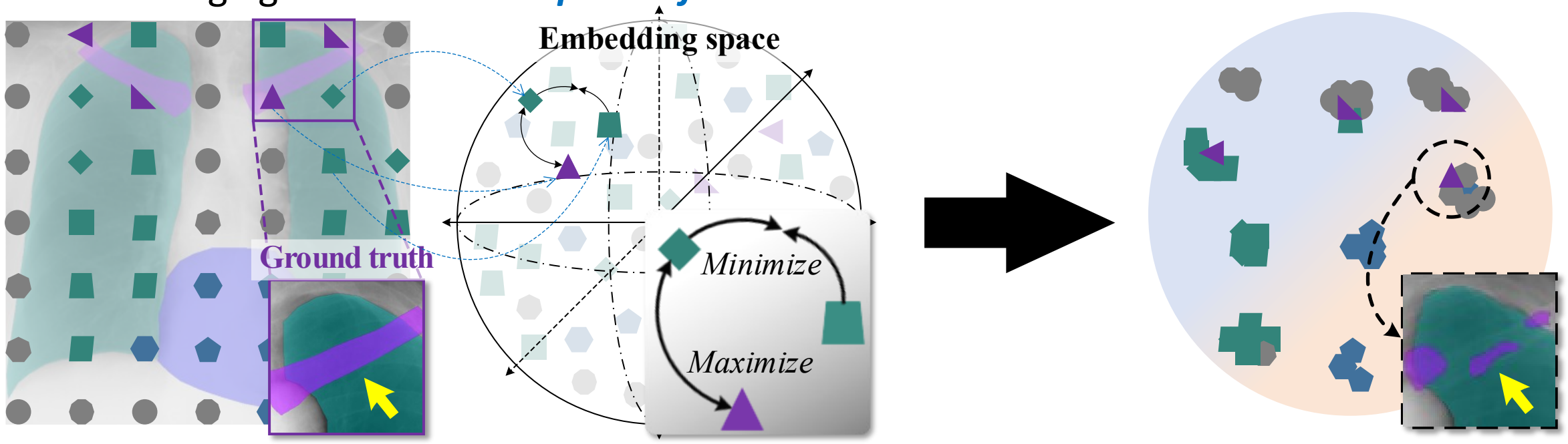
Medical images require pixel-wise representation to decouple underlying inner-scene semantics.

Problem – Over-dispersion problem

Pixel-wise features on image grid

Binary CL ***excessively disperses feature***

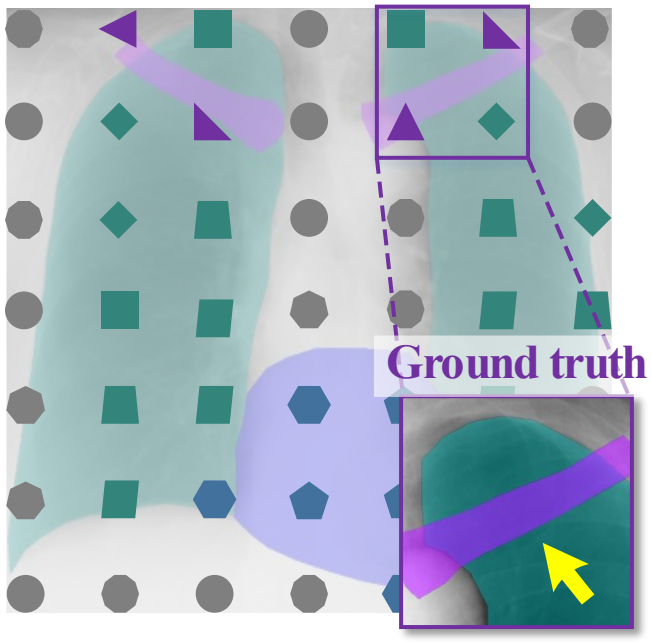
Over-dispersion *disrupts intra-class distribution*



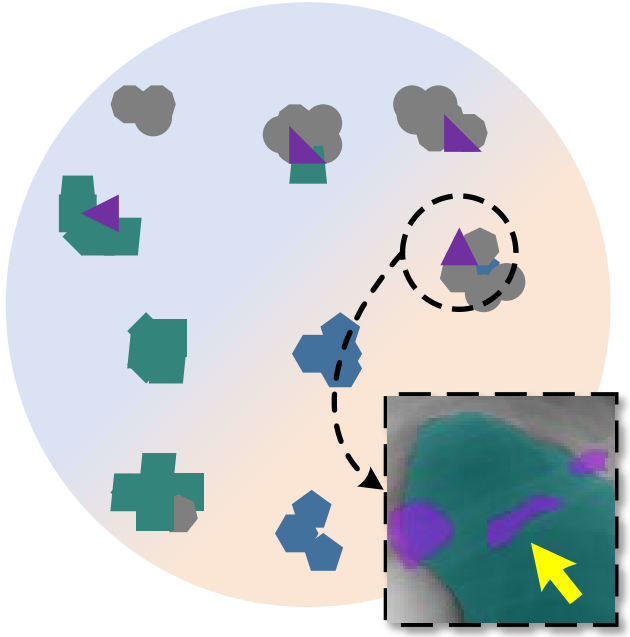
Over-dispersion problem: Excessive pursuit of dispersion disrupts intra-class distribution.

Problem – Over-dispersion problem

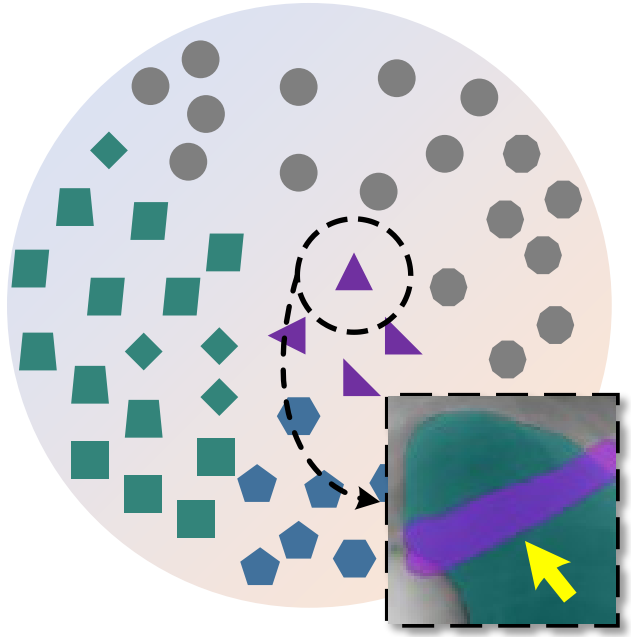
Pixel-wise features
on image grid



Over-dispersion *disrupts intra-class distribution*



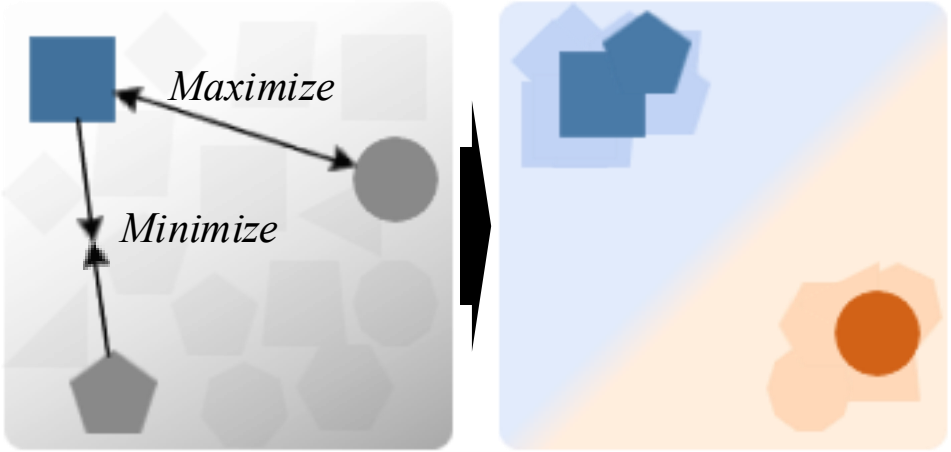
Ideal dispersion *preserves feature correlations*



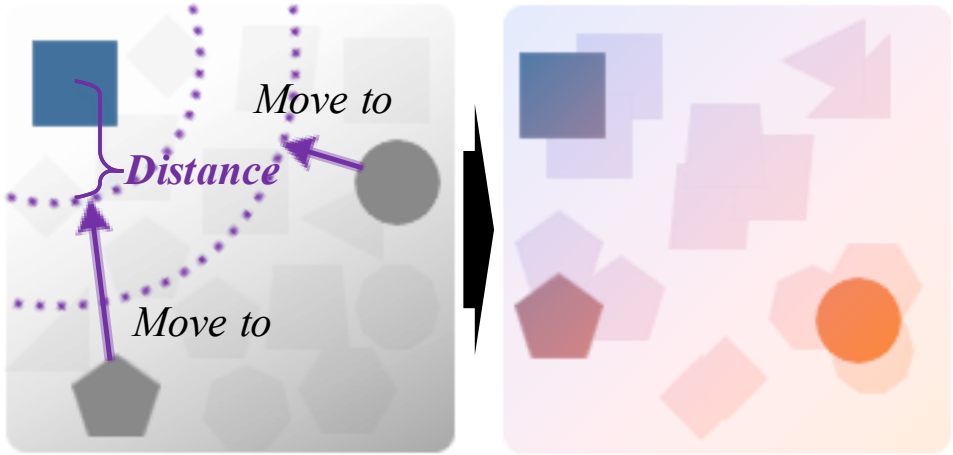
Over-dispersion problem: Excessive pursuit of dispersion disrupts intra-class distribution.

Motivation – Distance modeling

Binary CL ***excessively disperses feature causing over-dispersion***



Distance modeling ***quantifies dispersion degree***



Motivation: Predefine a ground truth to indicate the desired feature dispersion. (Unavailable)

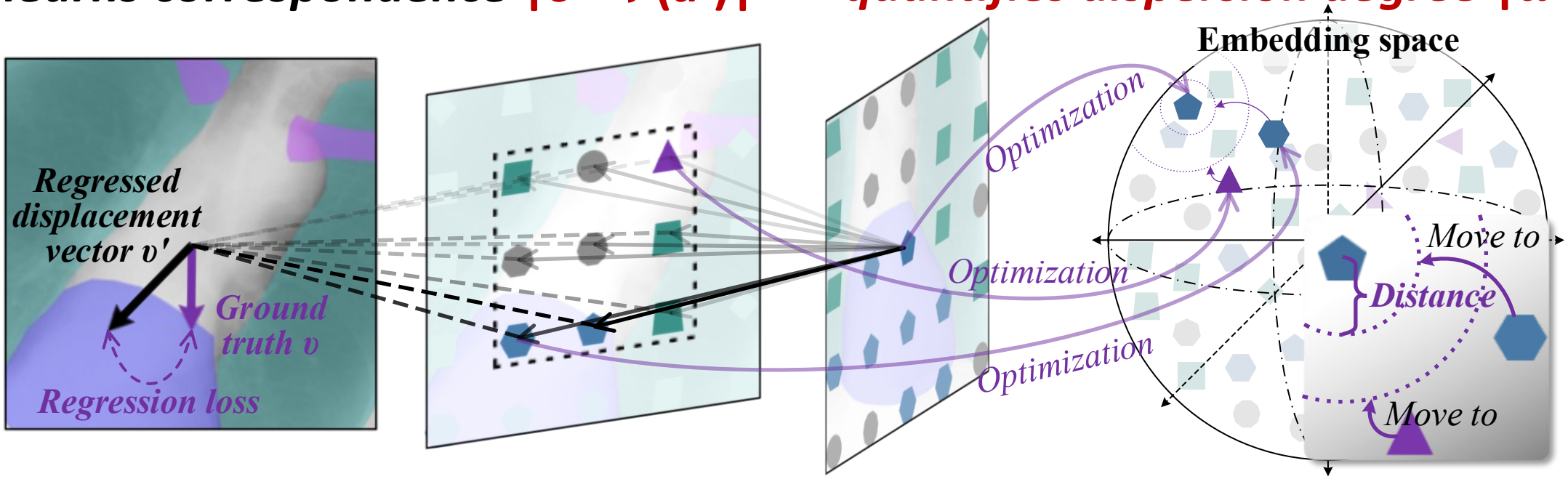
Motivation – Distance modeling as vector regression (Vector CL)

a) Vector regression in image space

learns correspondence $|v - \mathcal{V}(d')|$

b) Distance modeling in embedding space

quantifies dispersion degree $|\alpha - d'|$

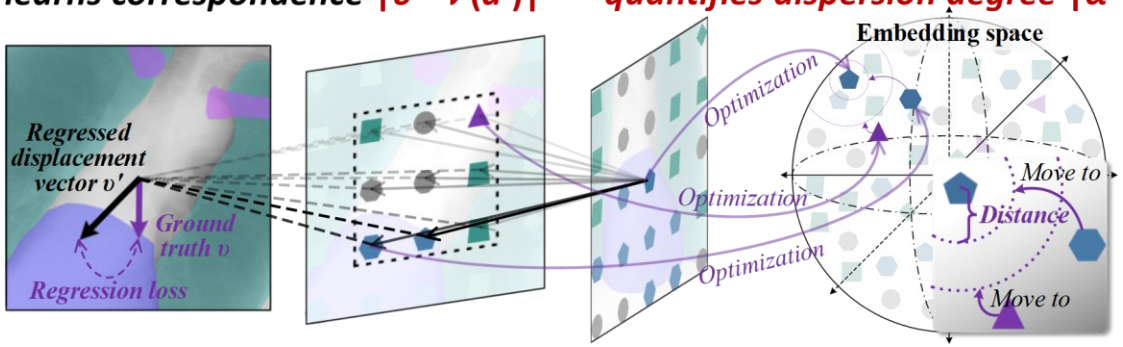


If the model wants to discover the correct correspondence, it must make potential same semantic features closer.

Formulation – Why does vector CL drive distance modeling?

a) Vector regression in image space
learns correspondence $|v - \mathcal{V}(d')|$

b) Distance modeling in embedding space
quantifies dispersion degree $|\alpha - d'|$



$$\textit{Dist. Mod.:} \sum_{i=0}^{I} |\alpha^i - d'^i| \to 0 \quad (2)$$

$$\Leftrightarrow \sum_{i=0}^{I} \mathbb{V}^i |\alpha^i - d'^i| \to 0 \quad (3)$$

$$\Rightarrow |\sum_{i=0}^{I} \mathbb{V}^i \alpha^i - \sum_{i=0}^{I} \mathbb{V}^i d'^i| \to 0 \quad (4)$$

$$\textit{Vec. Reg.:} \Leftrightarrow |v - \sum_{i=0}^{I} \mathbb{V}^i d'^i| \to 0. \quad (5)$$

1. Equ.2 is a distance modeling loss, which moves the features to a specific position with a certain distance.
2. Embed a template into the Equ.3, equivalent to Equ.2.
3. Expand Equ.3 for Equ.4, which is a one-way deduction.
4. Make $\sum_{i=0}^{I} \mathbb{V}^i \alpha^i = v$ for the vector regression loss in Equ.5.

Equ.4 captures the overall distance distribution rather than enforcing exact matches for each feature pair, enabling the model to accommodate varying feature biases across tasks.

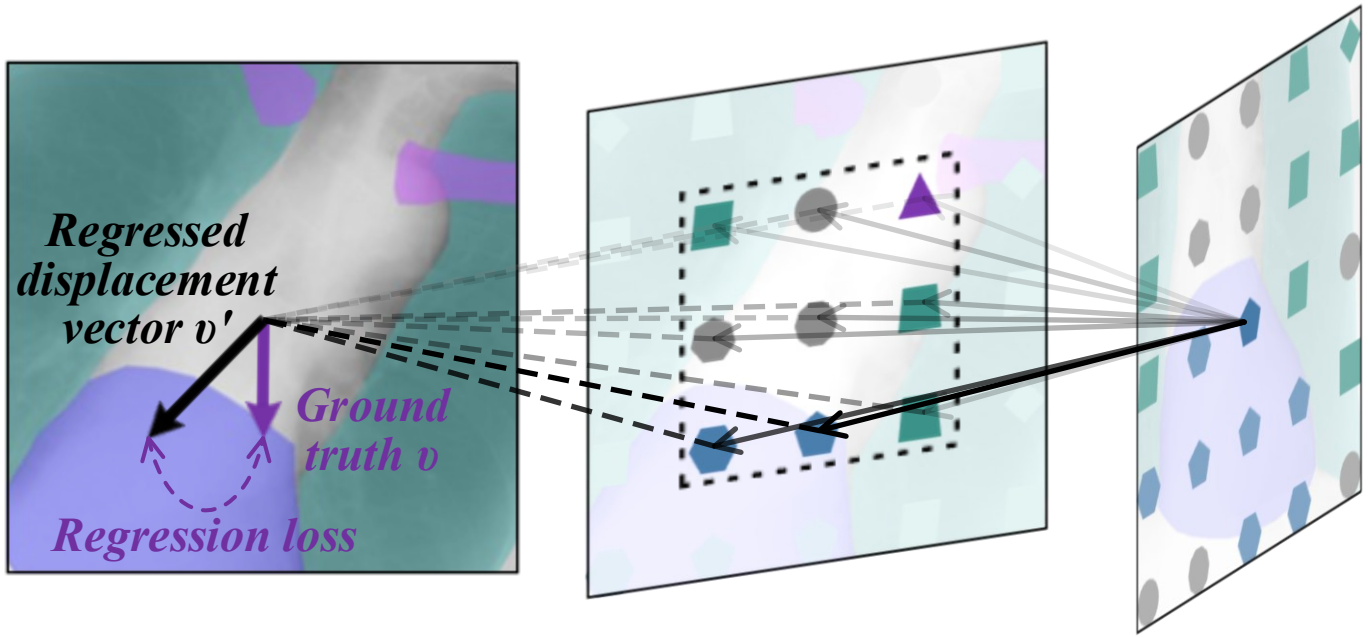
Questions – How to implement?

Q1: How to construct a self-learning paradigm with free ground truth vector υ that can be extended across diverse medical images?

Q2: How to formulate the function $\mathcal{V}$ to ensure a consistent optimization flow from vector regression to distance modeling?

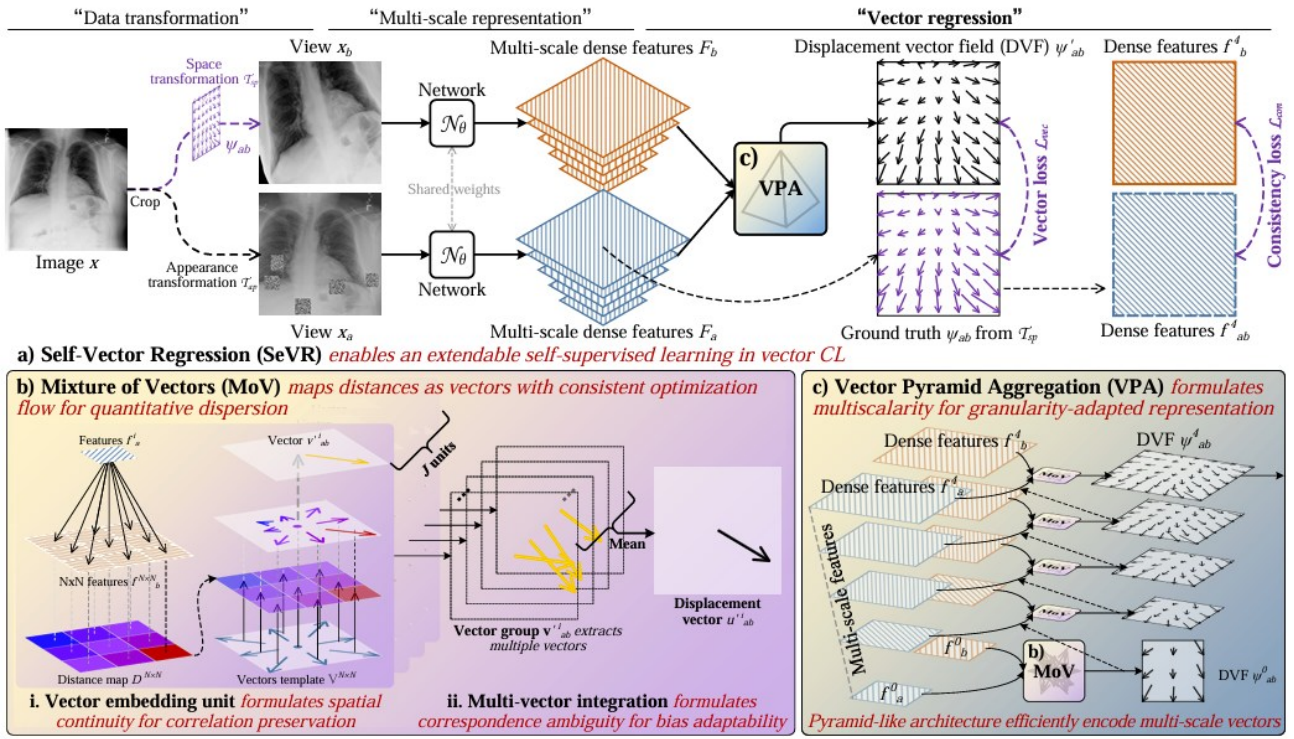
a) Vector regression in image space

learns correspondence $|\upsilon - \mathcal{V}(d')|$



Propose – COntrast in VEctor Regression (COVER) framework

- ✓ **For Q1 - Self-Vector Regression (SeVR):** constructs an extendable self-space transformation mechanism for vector CL with free ground truth vectors ***v.***
- ✓ **For Q2 - Mixture of Vectors (MoV):** formulates a mapping function with consistent optimization flow from vector regression to distance modeling with two properties.
- ✓ **For Q2 - Vector Pyramid Aggregation (VPA):** formulates multiscalarity of correspondence via stacking the MoV in a pyramid-like architecture.

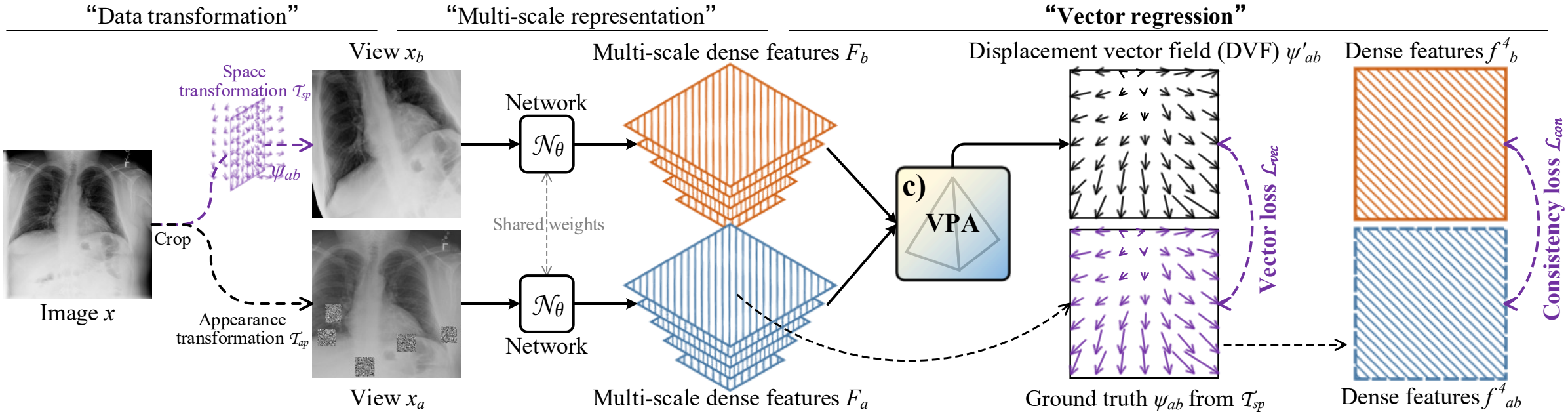


Innovation 1 – Self-Vector Regression (SeVR)

Self-Vector Regression (SeVR): constructs an extendable self-space transformation for vector CL with free ground truth vectors ***v.***

$$\min_{\theta} \mathbb{E}_{x\sim\mathcal{D}}[|\psi'_{ab} - \psi_{ab}|],$$

$$\text{where } \psi'_{ab} = \mathcal{V}(F_a, F_b),\ \psi_{ab} \sim \mathcal{T}_{sp},\ \psi_{ab}(x_a) \equiv x_b.$$

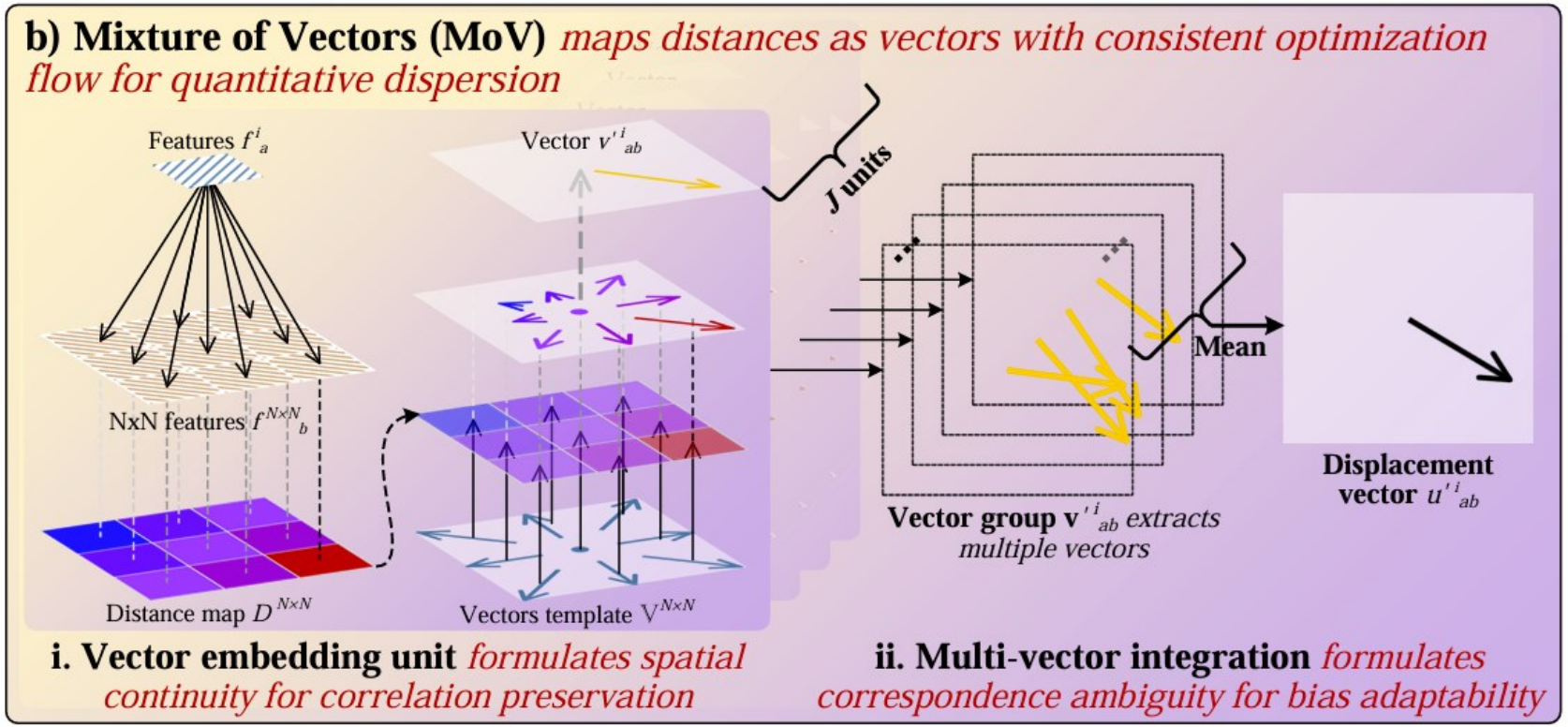


Advantages: Extendable to a wide variety of medical images for large-scale training simplifying data preparation and enhancing generality.

Innovation 2 – Self-Vector Regression (SeVR)

Mixture of Vectors (MoV): formulates a mapping function with consistent optimization flow from vector regression to distance modeling with two properties.

$$v'^{i}_{ab} = \mathcal{U}(f^{i}_{a}, f^{N\times N}_{b}) = \text{softmax}(\frac{f^{i}_{a} f^{\top N\times N}_{b}}{\tau})\mathbb{V}^{N\times N}, \quad (7)$$

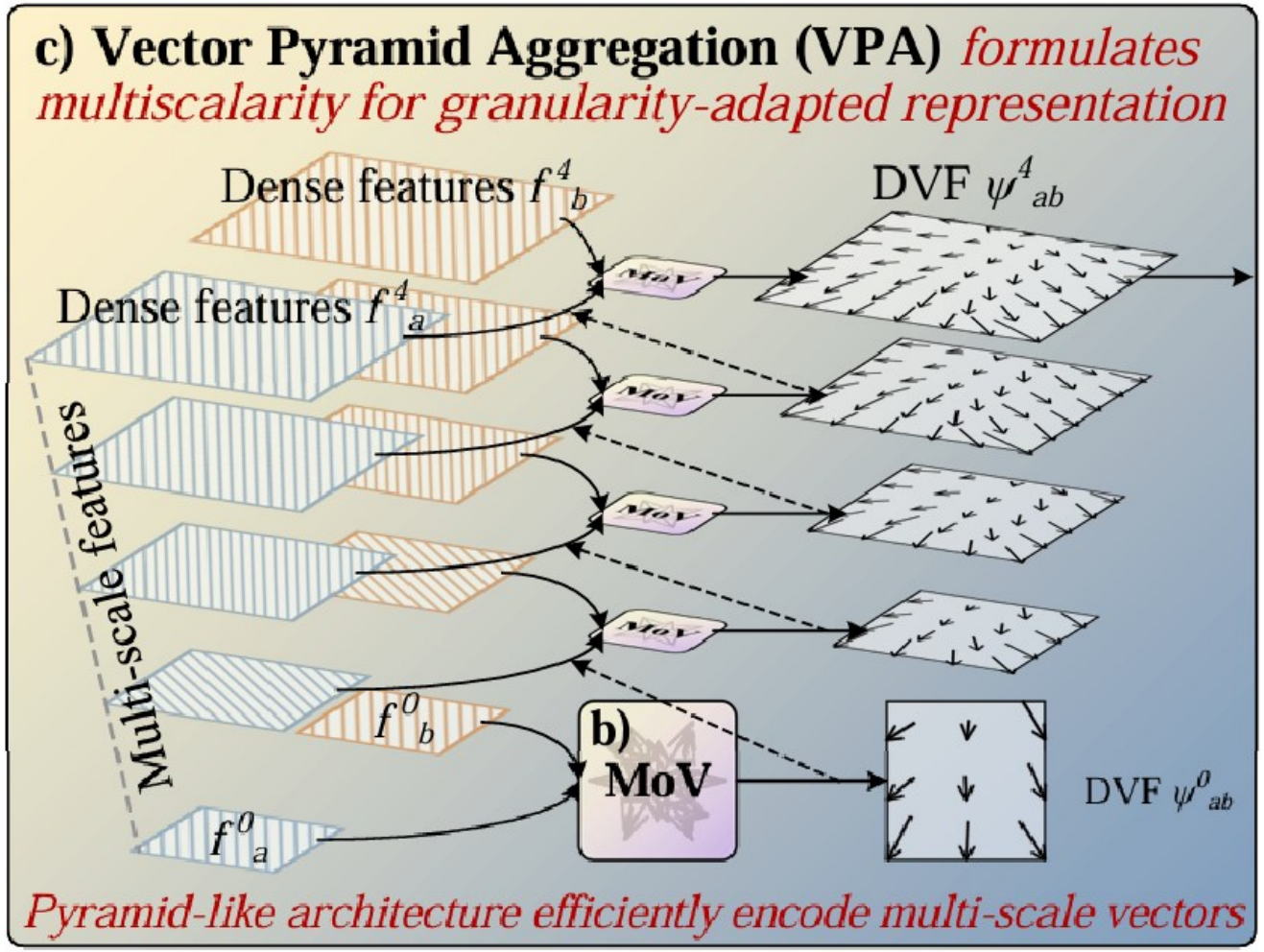


Advantages:

1. Feature correlation preservation: vector template describes continuous spatial relationship, avoid the artificial division.

2. Bias adaptability: MVI models the ambiguity to represent diverse feature concerns, enhancing bias adaptability in general applications.

Innovation 3 – Vector Pyramid Aggregation (VPA)



Vector Pyramid Aggregation (VPA): formulates multiscalarity of correspondence via stacking the MoV in a pyramid-like architecture.

$$\psi'_{ab} = \mathcal{V}(F_a, F_b) = H(\{\psi'^l_{ab}\}_{l=0}^{L}), \text{ where} \quad (8)$$
$$\psi'^0_{ab} = \mathcal{M}(f^0_a, f^0_b)$$
$$\psi'^l_{ab} = \mathcal{M}(\psi'^{l-1}_{ab}(f^l_a), f^l_b) \odot \psi'^{l-1}_{ab},\ l = 1, 2, ..., L-1,$$

Advantages:

1. Low computational cost: It enables the mapping function in a small receptive field at each level for a large whole receptive field reducing the computation.

2. Multi-scale representation: It learns the multi-scale features with multiple semantic granularities, improving the granularity adaptability.

Comparison study – Comparison setting

Dataset	Type	Num	D	P	Task
CANDI [19]	3D T1 brain MRI	103	√		S
FeTA21 [30]	3D T2 brain MRI	80	√		S
SCR [35]	2D chest X-ray	247	√		S
KiPA22 [12]	3D kidney CT	130	√		S
FIVES [18]	2D fundus	800	√		S
PDCXR [20]	2D chest X-ray	5,956	√		C
STOIC [31]	3D chest CT	2,000	√		C
ChestX-ray8 [36]	2D chest X-ray	112,120		√	-
PPMI (T1) [25]	3D T1 brain MRI	837		√	-

Table 1. A Total of 9 publicly available datasets are involved in this paper for the experiments, achieving great reproducibility. The "D" and "P" mean the datasets are used for downstream tasks and pretraining tasks. The "S" and "C" are the segmentation and classification tasks.

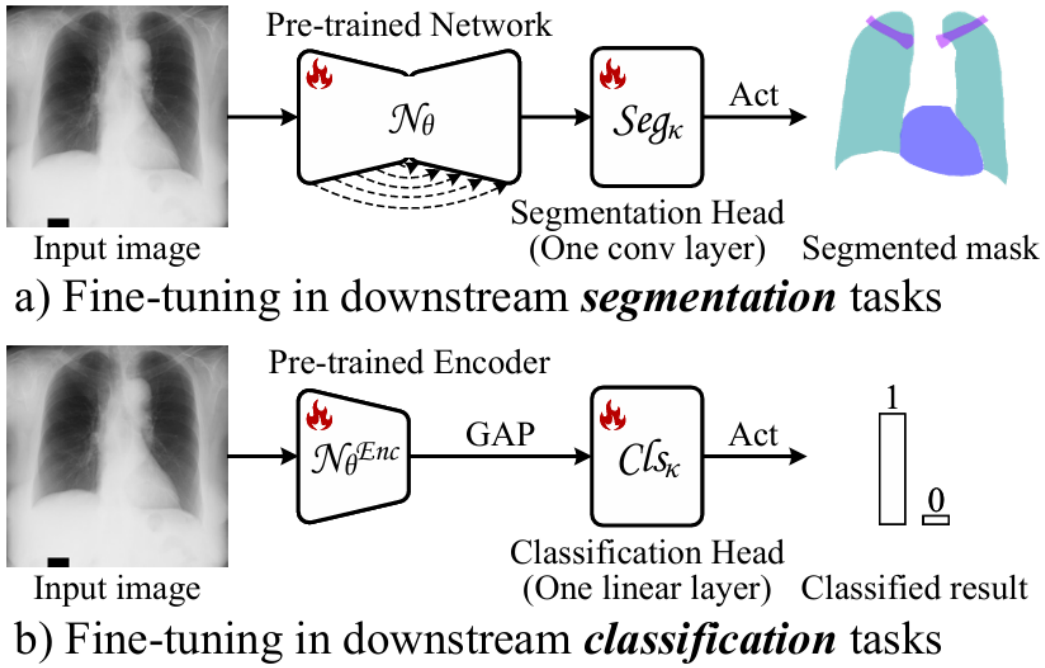


Figure 2. The detailed implementations in our downstream tasks, including the a) segmentation and b) classification.

Evaluate our method across 8 tasks, spanning 2 dimensions and 4 modalities with 2 kinds of downstream tasks.

Comparison study – SOTA performance compared with the existing methods

Type	Methods	a) 2D evaluation pretrained on [59]				b) 3D evaluation pretrained on [31]				AVG
		$SCR_S^{25\%}$	$PDCXR_C$	$KiPA22_S^{2D}$	$FIVES_S$	$CANDI_S$	$FeTA21_S$	$KiPA22_S^{3D}$	$STOIC_C$	*Score %*
-	Scratch	81.8	90.4	74.1	79.4	84.0	56.9	72.4	72.0	76.4
GL	Denosing [54]	$83.9_{(+1.9)}$	$92.0_{(+1.6)}$	$60.3_{(-13.8)}$	$77.8_{(-1.6)}$	$83.7_{(-0.3)}$	$52.9_{(-4.0)}$	$70.0_{(-2.4)}$	$65.9_{(-6.1)}$	$73.3_{(-3.1)}$
	In-painting [38]	$85.1_{(+3.3)}$	$93.9_{(+3.5)}$	$64.4_{(-9.7)}$	$78.9_{(-0.5)}$	$88.5_{(+4.5)}$	$54.4_{(-2.5)}$	$69.7_{(-2.7)}$	$67.2_{(-4.8)}$	$75.3_{(-1.1)}$
	Models Genesis [69]	$86.1_{(+4.3)}$	$92.6_{(+2.2)}$	$66.6_{(-7.5)}$	$79.6_{(+0.2)}$	$88.7_{(+4.7)}$	$55.8_{(-1.1)}$	$75.8_{(+3.4)}$	$75.3_{(+3.3)}$	$77.6_{(+1.2)}$
	Rotation [28]	$80.5_{(-1.3)}$	$89.9_{(-0.5)}$	$69.7_{(-4.4)}$	$80.3_{(+0.9)}$	$89.4_{(+5.4)}$	$58.7_{(+1.8)}$	$77.4_{(+5.0)}$	$68.8_{(-3.2)}$	$76.8_{(+0.4)}$
BCL	SimSiam [8]	$87.2_{(+5.4)}$	$92.2_{(+1.8)}$	$72.6_{(-1.5)}$	$84.3_{(+4.9)}$	$87.3_{(+3.3)}$	$58.7_{(+1.8)}$	$83.8_{(+11.4)}$	$69.5_{(-2.5)}$	$79.5_{(+3.1)}$
	BYOL [14]	$89.4_{(+7.6)}$	$86.6_{(-3.8)}$	$74.1_{(0)}$	$83.3_{(+3.9)}$	$89.7_{(+5.7)}$	$59.2_{(+2.3)}$	$83.6_{(+11.2)}$	$74.0_{(+2.0)}$	$80.0_{(+5.4)}$
	SimCLR [7]	$89.0_{(+7.2)}$	$94.7_{(+4.3)}$	$74.4_{(+0.3)}$	$84.5_{(+5.1)}$	$89.2_{(+5.2)}$	$53.4_{(-3.5)}$	$78.9_{(+6.5)}$	$60.7_{(-11.3)}$	$78.1_{(+1.7)}$
	MoCov2 [9]	$84.3_{(+2.5)}$	$93.2_{(+2.8)}$	$69.6_{(-4.5)}$	$80.7_{(+1.3)}$	$89.7_{(+5.7)}$	$61.5_{(+4.6)}$	$78.0_{(+5.6)}$	$74.8_{(+2.8)}$	$79.0_{(+2.6)}$
	DeepCluster [4]	$84.0_{(+2.2)}$	$93.3_{(+2.9)}$	$72.7_{(-1.4)}$	$81.6_{(+2.2)}$	$89.8_{(+5.8)}$	$57.4_{(+0.5)}$	$79.7_{(+7.3)}$	$65.6_{(-6.4)}$	$78.0_{(+1.6)}$
DBCL	VADeR [35]	$85.2_{(+3.4)}$	$92.8_{(+2.4)}$	$62.8_{(-11.3)}$	$78.9_{(-0.5)}$	$87.4_{(+3.4)}$	$43.1_{(-13.8)}$	$72.1_{(-0.3)}$	$73.2_{(+1.2)}$	$74.4_{(-2.0)}$
	DenseCL [60]	$85.0_{(+3.2)}$	$92.4_{(+2.0)}$	$70.8_{(-3.3)}$	$79.2_{(-0.2)}$	$87.7_{(+3.7)}$	$43.7_{(-13.2)}$	$74.0_{(+1.6)}$	$58.8_{(-13.2)}$	$74.0_{(-2.4)}$
	SetSim [61]	$85.2_{(+3.4)}$	$93.9_{(+3.5)}$	$70.8_{(-3.3)}$	$80.1_{(+0.7)}$	$88.4_{(+4.4)}$	$58.7_{(+1.8)}$	$73.5_{(+1.1)}$	$60.1_{(-11.9)}$	$76.3_{(-0.1)}$
	DSC-PM [29]	$90.5_{(+8.7)}$	$91.8_{(+1.4)}$	$77.2_{(+3.1)}$	$83.8_{(+4.4)}$	$88.5_{(+4.5)}$	$52.2_{(-4.7)}$	$79.0_{(+6.6)}$	$59.5_{(-12.5)}$	$77.8_{(+1.2)}$
	PixPro [64]	$91.5_{(+9.7)}$	$93.0_{(+2.6)}$	$73.6_{(-0.5)}$	$84.3_{(+4.9)}$	$89.9_{(+5.9)}$	$60.7_{(+3.8)}$	$80.0_{(+7.6)}$	$75.1_{(+3.1)}$	$81.0_{(+4.6)}$
	Chaitanya et al. [5]	$87.3_{(+5.5)}$	$90.0_{(-0.4)}$	$76.5_{(+2.4)}$	$84.9_{(+5.5)}$	$87.4_{(+3.4)}$	$53.4_{(-3.5)}$	$70.7_{(-1.7)}$	$67.8_{(-4.2)}$	$77.3_{(+0.9)}$
VR	GVSL [19]	$89.7_{(+7.9)}$	$92.5_{(+2.1)}$	$78.9_{(+4.8)}$	$86.2_{(+6.8)}$	$89.1_{(+5.1)}$	$62.6_{(+5.7)}$	$84.3_{(+11.9)}$	$75.4_{(+3.4)}$	$82.3_{(+5.9)}$
	GEMINI [22]	$92.4_{(+10.6)}$	$92.9_{(+2.5)}$	$79.1_{(+5.0)}$	$85.3_{(+5.9)}$	**$90.0_{(+6.0)}$**	$61.7_{(+4.8)}$	$85.0_{(+12.6)}$	$79.5_{(+7.5)}$	$83.2_{(+6.8)}$
VCL	**COVER (Ours)**	**$94.0_{(+12.2)}$**	**$95.9_{(+5.5)}$**	**$80.0_{(+5.9)}$**	**$87.2_{(+7.8)}$**	$89.9_{(+5.9)}$	**$63.6_{(6.7)}$**	**$85.2_{(+12.8)}$**	**$80.4_{(+8.4)}$**	**$84.5_{(+8.1)}$**

- **Observation:** Significantly improve the performance across scales and scenes, increasing over 8% compared with the “Scratch”.
- **Conclusion:** Our proposed COVER has achieved SOTA performance compared with the existing methods.

Ablation study – Component ablation

	base. $\mathcal{L}_{con}$	...+VEU (SeVR)	...+VPA	...+MVI
DSC%	91.8	92.9	93.4	94.0

- **Observation:** The proposed modules are gradually added, and the performance of the model is gradually improved.
- **Conclusion:** The proposed modules all have gain effects on the performance.

Ablation study – Hyper-parameter ablation

Receptive field in VEU: *N*

$N =$	3×3	5×5	7×7	9×9
DSC%	44.9	48.4	54.8	35.1

Amount of VEUs in MoV: *J*

$J =$	$[2, 2, 2, 1, 1]$	$[4, 4, 4, 1, 1]$	$[8, 8, 8, 2, 2]$	$[12, 12, 12, 3, 3]$
DSC%	55.3	56.3	54.8	48.6

- **Observation for *N*:** Increase and then decrease.
- **Explanation for *N*:** A too large *N* will introduce more ambiguous semantics, misleading the correspondence.

- ***Observation for J:*** Increase and then decrease.
- **Explanation of *J*:** Too many vectors will smooth the optimization, weakening the discrimination of features.

Model analysis – Implementation analysis

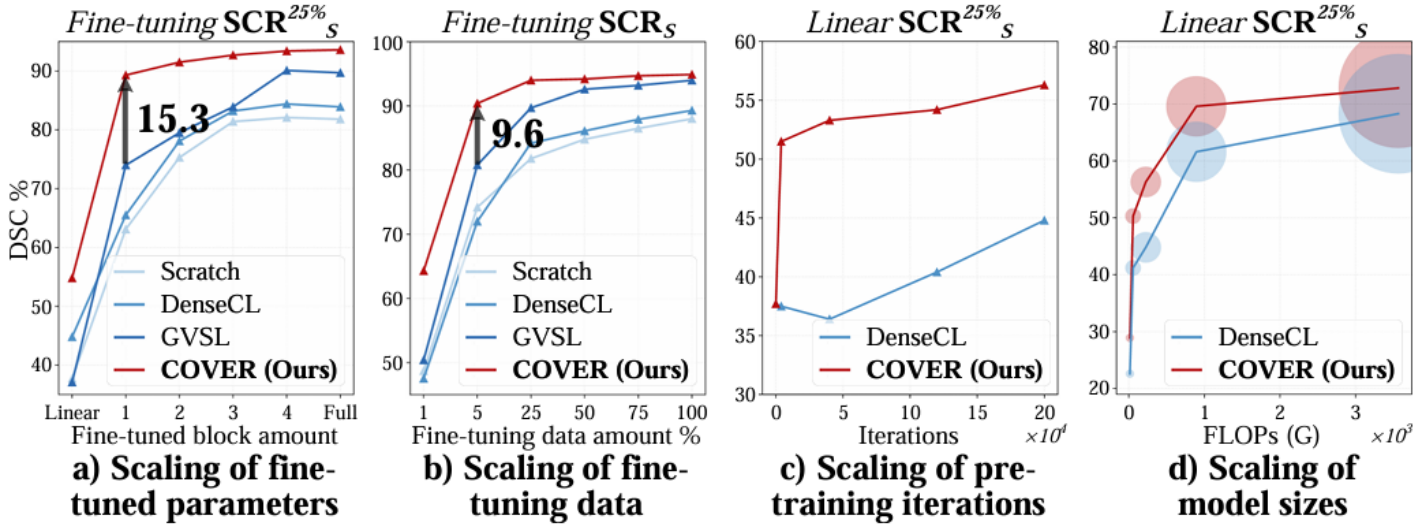


Figure 4. **Implementation analysis:** Our COVER has great properties on the scaling of a) fine-tuned parameters, b) fine-tuning data, c) pretraining iterations, and d) model sizes.

- **Parameters:** Our COVER effectively reduces the fine-tuned parameters.
- **Finetuning data:** Our COVER effectively reduces the fine-tuning data requirement.
- **Pre-training iterations:** With the progress of the training iterations, the performance of our COVER gradually improves and eventually tends to be flat.
- **Mode size:** With the enlarging of model size, the larger capacity enables our COVER to gain more powerful representability.

Model analysis – Convergence analysis

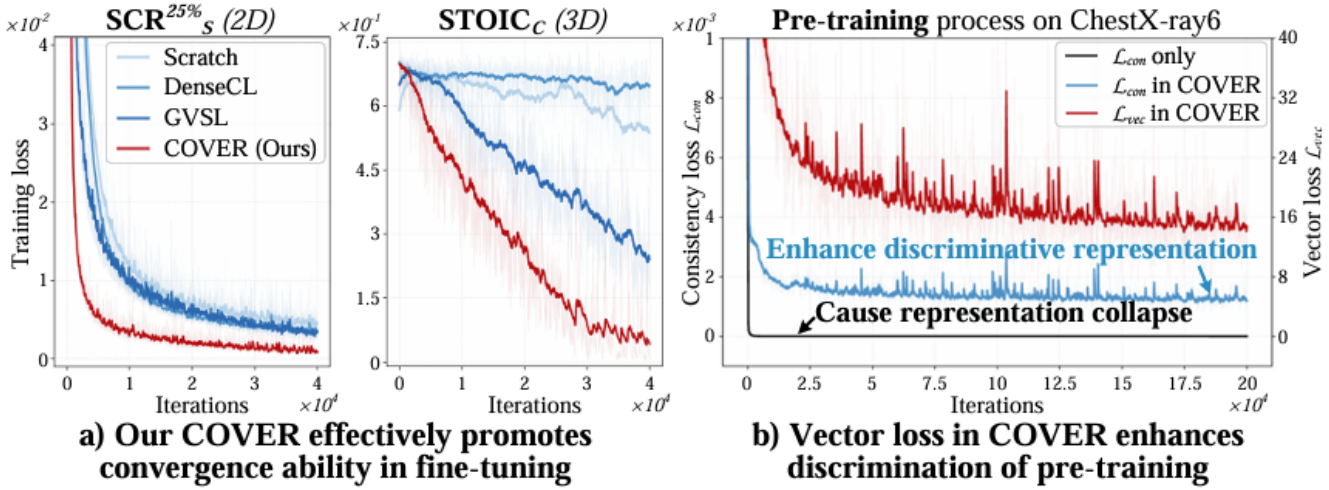
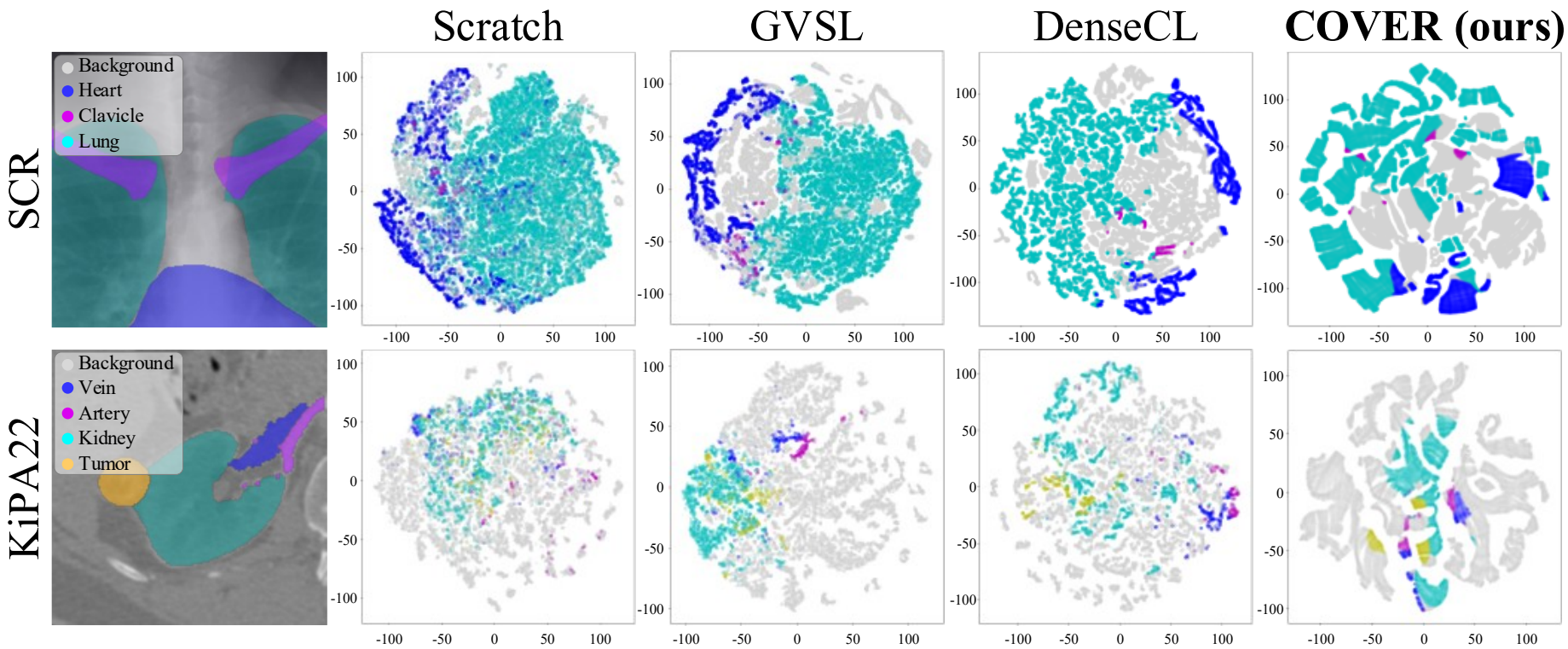


Figure 5. **Convergence analysis:** a) Our COVER is able to promote convergence ability in fine-tuning. b) In the pretraining, the consistency learning causes representation collapse, and when adding our vector regression, the discrimination is enhanced.

- **Fine-tuning:** The network pre-trained by our COVER exhibits superior convergence ability.
- **Pre-training:** COVER improves the discrimination of the representation.

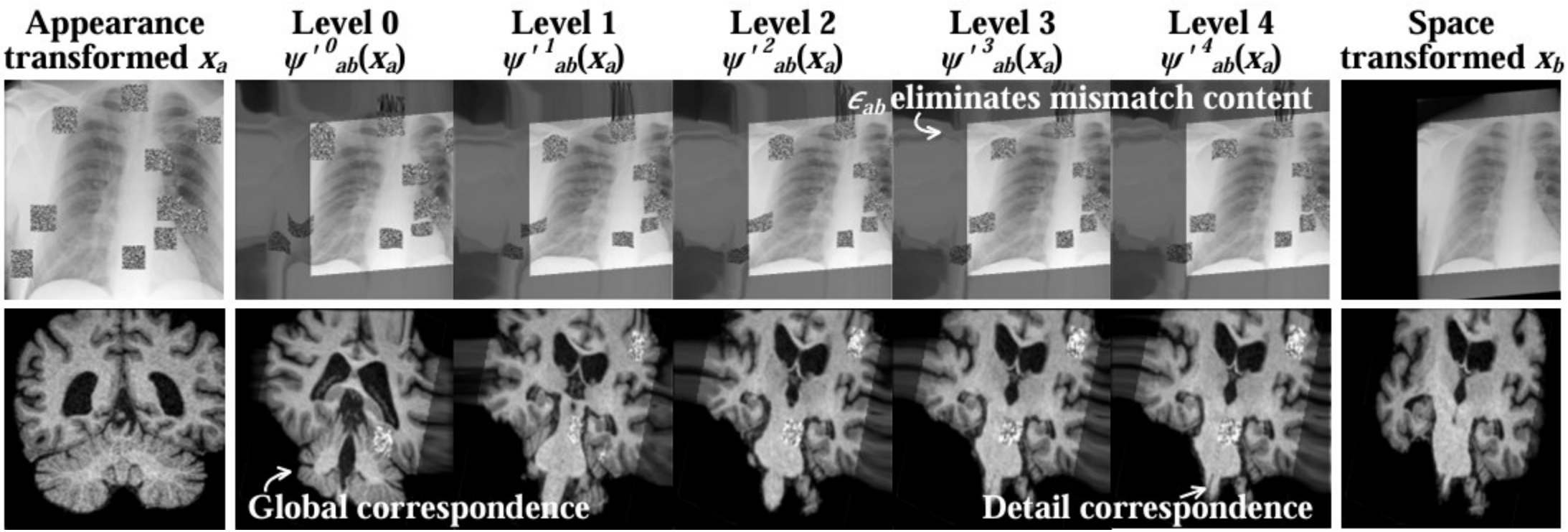
Model analysis – Distributed clusters from COVER



Two advantages:

- Continuous feature distribution preserves feature correlations.
- Effective aggregation provides distinct representation.

Model analysis – Visualization of multi-scale vectors



- In level 0, the DVF predicted from global features can align the images on the whole, driving the learning of global representation. With the expansion of the scales, the correspondences are gradually refined so the details between the images are aligned.

Discussion – Theoretical Foundation

Foundation of Rademacher Complexity

Feature distance: $\langle f(x)_i, f(x)_j \rangle \leq \delta, \quad \forall j \in \mathcal{N}(i),$

Range of distance: $\langle f(x)_i, f(x)_j \rangle \rightarrow \text{anywhere within} [-\Delta, \Delta],$

Upper bound of δ: $\delta = \max_{j,k \in \mathcal{N}(i)} |\langle f(x)_i, f(x)_j \rangle - \langle f(x)_i, f(x)_k \rangle|.$

Local Rademacher complexity: $\mathfrak{R}_{n,\text{local}}(\mathcal{F}) \leq \left(\frac{\delta}{\Delta}\right) \mathfrak{R}_n(\mathcal{F}),$

Generalization error: $R(f) \leq \hat{R}(f) + 2\mathfrak{R}_{n,\text{local}}(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right),$

$$\Rightarrow \quad R(f) \leq \hat{R}(f) + 2\left(\frac{\delta}{\Delta}\right) \mathfrak{R}_n(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right).$$

Over-Dispersion Problem in Binary CL

Distance in BCL: $\delta_{BCL} \approx \Delta.$

BCL's Generalization error: $R_{BCL}(f) \leq \hat{R}(f) + 2\mathfrak{R}_n(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right),$

Discussion – Theoretical Foundation

Over-Dispersion Problem in Binary CL

Distance in BCL: $\delta_{BCL} \approx \Delta.$

BCL's Generalization error: $R_{BCL}(f) \leq \hat{R}(f) + 2\mathfrak{R}_n(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right),$

Vector contrastive learning

VCL loss: $\mathcal{L}_{VCL} = \left\| v - \sum_{j=0}^{\mathcal{N}(i)} \mathbb{V}^j \frac{e^{\langle f(x)_i, f(x)_j \rangle / \tau}}{\sum_j^{\mathcal{N}(i)} e^{\langle f(x)_i \cdot f(x)_j \rangle / \tau}} \right\|$

Hypothesis: $v = \sum_j^{\mathcal{N}(i)} \alpha_j \mathbb{V}^j \ (\alpha_j \geq 0, \sum_j^{\mathcal{N}(i)} \alpha_j = 1),$

Inference: $\frac{e^{\langle f(x)_i, f(x)_j \rangle / \tau}}{Z} \approx \alpha_j,$

$$\Rightarrow \quad Z = \sum_j^{\mathcal{N}(i)} e^{\langle f(x)_i, f(x)_j \rangle}.$$

$$\Rightarrow \quad \langle f(x_i), f(x_j) \rangle \approx \tau \log \alpha_j + \tau \log Z.$$

Distance in VCL:

$$\begin{aligned} \delta_{VCL} &= \max_{i,k} |\langle f(x_i), f(x_j) \rangle - \langle f(x_i), f(x_k) \rangle| \\ &= \max_{i,k} |\tau \log \alpha_j + \tau \log Z - \tau \log \alpha_k - \tau \log Z| \\ &= \tau \max_{i,k} |\log \frac{\alpha_j}{\alpha_k}|. \end{aligned} \tag{15}$$

Inference: $\delta_{VCL} \leq \tau \log \frac{1}{\alpha_{\min}} \ll \Delta.$

VCL's Generalization error:

$$R_{VCL}(f) \leq \hat{R}(f) + 2\left(\frac{\tau \log \frac{1}{\alpha_{\min}}}{\Delta}\right) \mathfrak{R}_n(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right)$$

Discussion – Theoretical Foundation

- **BCL's Generalization error:**

$$R_{BCL}(f) \leq \hat{R}(f) + 2\mathfrak{R}_n(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right),$$

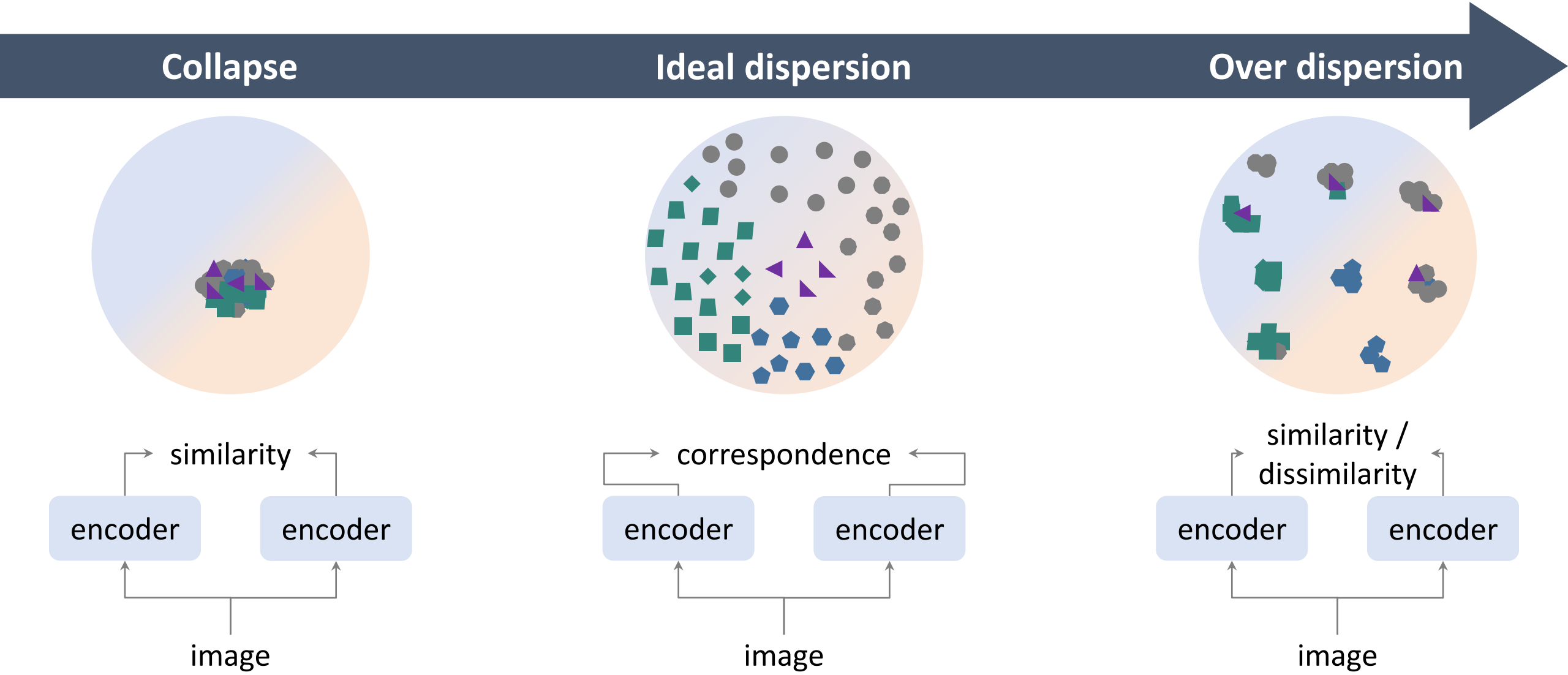
- **VCL's Generalization error:**

$$R_{VCL}(f) \leq \hat{R}(f) + 2\left(\frac{\tau \log \frac{1}{\alpha_{\min}}}{\Delta}\right)\mathfrak{R}_n(\mathcal{F}) + O\left(\sqrt{\frac{\log(1/\varepsilon)}{nN}}\right)$$

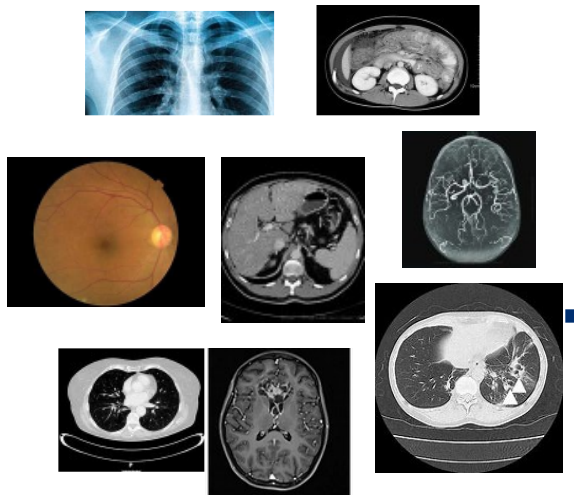
s.t. $\delta_{VCL} \leq \tau \log \frac{1}{\alpha_{\min}} \ll \Delta.$

$\Rightarrow$ **Vector CL makes a tighter generalization bound.**

Future work 1 – Vector contrastive representation learning



Future work 2 – Large pixel-wise medical vision model



Data

COVER

- **Segmentation**
- **Classification**
- **Regression**
- **Detection**
- **Registration**
- **NICT imaging**

Thanks for listening

Q & A