Supplementary materials for the paper "VNE: An Effective Method for Improving Deep Representation by Manipulating Eigenvalue Distribution"

A. A Brief Introduction to Quantum Theory

A classic bit can be either 0 or 1. In quantum theory [67, 88], a *qubit* is a quantum extension of the classic bit, and it can be in state $|0\rangle$, state $|1\rangle$, or any linear combination (superposition state) of the two as $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$, where $|a|^2 + |b|^2 = 1$.

Dirac notation and basic concepts: Dirac notation is used in quantum theory [24]. For a state $|\psi\rangle$, ψ should be understood as the name or label of the state. Because linear algebra provides the mathematical foundation of quantum theory, vector notation is adopted. For instance, in the simple example of $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$, $|\psi\rangle$ can be expressed as $|\psi\rangle = [a, b]^T$ where the interpretation should be state $|\psi\rangle$ can be 0 with probability $|a|^2$ and 1 with probability $|b|^2$ (therefore $|a|^2 + |b|^2 = 1$). Here, the *ket* vector $|\psi\rangle$ is the Dirac notation for a column vector in a Hilbert space $\mathcal{H}$. To represent a row vector, the *bra* vector $\langle\psi|$ is used, as in $\langle\psi| = [a, b]$. An inner product or *braket* is represented as $\langle\psi|\phi\rangle$ and an outer product or *ketbra* is represented as $|\psi\rangle\langle\phi|$.

A *composite quantum state* of n qubits can be represented as a vector of size 2^n (e.g., a single-qubit state is represented as a vector of size two). For example, a quantum state of two separable single-qubit states can be represented as

$$\begin{aligned} |\psi\rangle \otimes |\phi\rangle &= |\psi\rangle\,|\phi\rangle = |\psi\phi\rangle \\ &= [a, b]^T \otimes [c, d]^T = [ac, ad, bc, bd]^T \end{aligned} \tag{8}$$

in which $|ac|^2$, $|ad|^2$, $|bc|^2$, and $|bd|^2$ represent the probability of $|\psi\phi\rangle$ being $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$, respectively. In d-dimensional quantum system, a quantum state is on the unit hypersphere in a Hilbert space $\mathcal{H}$.

A state can be either *pure* or *mixed*. In the simple example, $|0\rangle = [1, 0]^T$ and $|1\rangle = [0, 1]^T$ form the *computational basis states*, and they are pure states. Any superposition of the two, $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$, is also a pure state because it corresponds to a single vector with a probabilistic distribution over the basis states. By contrast, a mixed state is a probabilistic mixture of a set of pure states. Note that a pure state already has a probabilistic interpretation over the basis states and a mixed state has an additional level of probabilistic interpretation over a set of such pure states. In this case, we are considering a state that is not completely known but is an ensemble of pure states $\{|\psi_i\rangle\}$ with respective probabilities $\{p_i\}$. The full information of a mixed state cannot be represented as a vector, and the notion of the density operator (also called density matrix) is required.

Definition 1 (Density operator [67])**.** A density operator is defined as below.

$$\rho \triangleq \sum_i p_i\,|\psi_i\rangle\langle\psi_i|\,. \tag{9}$$

Density operator ρ satisfies $\rho \geq 0$ and $tr(\rho) = 1$. In addition, $\rho = \rho^2$ and $rank(\rho) = 1$ are satisfied for pure states and $tr(\rho^2) < 1$ is satisfied for mixed states. The density operator provides a convenient way to describe the uncertainty or probability distribution of a quantum system. According to Gleason's theorem [31], the probability of a state $|\psi_i\rangle$ in the system with ρ is given by $tr(\rho\,|\psi_i\rangle\langle\psi_i|)$.

While quantum theory encompasses a broad scope of subjects, quantum information theory or quantum Shannon theory is a sub-field that focuses on the quantum equivalent of Shannon information theory [88]. Among the extensive results, we utilize the basic concepts of *von Neumann entropy* (also called quantum entropy). While Shannon entropy is calculated for a classical probability distribution, von Neumann entropy is calculated for a density operator ρ [67], a positive semi-definite hermitian matrix in a Hilbert space $\mathcal{H}$ with the trace value of one. Similar to Shannon information theory, it measures the uncertainty associated with a quantum system.

Definition 2 (von Neumann entropy [67])**.** The von Neumann entropy (quantum entropy) of a quantum state with density operator ρ is defined as

$$S(\rho) \triangleq -tr(\rho \log \rho) = -\sum_j \lambda_j \log \lambda_j, \tag{10}$$

where $\{\lambda_j\}$ are the eigenvalues of ρ.

B. Proofs of Theorems

Lemma 1. *For given $p_i \geq 0$ and $\sum_{i=1}^n p_i = 1$, the entropy function $H(p_1, ..., p_n) = -\sum_{i=1}^n p_i \log p_i$ is strictly concave and is upper-bounded by $\log n$ as follows,*

$$\log n = H(1/n, ..., 1/n) \geq H(p_1, ..., p_n) \geq 0. \tag{11}$$

Proof. Refer to Section D.1 in [61]. □

Lemma 2. *The KL Divergence for two zero-mean d-dimensional multivariate Gaussian distributions can be derived as follows,*

$$\begin{aligned} &D_{\mathrm{KL}}(\mathcal{N}(0, \boldsymbol{\Sigma}_1) \| \mathcal{N}(0, \boldsymbol{\Sigma}_2)) \\ &= \frac{1}{2}\left[tr(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1) - d + \log \frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|}\right]. \end{aligned} \tag{12}$$

Proof. Refer to Section 9 in [26]. □

Theorem 1 (Rank and VNE). *For a given representation autocorrelation $\mathcal{C}_{auto} = \boldsymbol{H}^T\boldsymbol{H}/N \in \mathbb{R}^{d\times d}$ of rank k ($\leq d$),*

$$log(rank(\mathcal{C}_{auto})) \geq S(\mathcal{C}_{auto}), \tag{13}$$

where equality holds iff the eigenvalues of $\mathcal{C}_{auto}$ are uniformly distributed with $\forall_{j=1}^{k}\lambda_j = 1/k$ and $\forall_{j=k+1}^{d}\lambda_j = 0$.

Proof.

$$\begin{aligned}
\log(\text{rank}(\mathcal{C}_{\text{auto}})) &= \log(k) && (14)\\
&\geq H(\lambda_1, ..., \lambda_k) \text{ (by Lemma 1)} && (15)\\
&= -\sum_{j=1}^{k}\lambda_j \log\lambda_j && (16)\\
&= -\sum_{j=1}^{d}\lambda_j \log\lambda_j && (17)\\
&= S(\mathcal{C}_{\text{auto}}). && (18)
\end{aligned}$$

By Lemma 1, the inequality (15) holds with equality if and only if $\forall_{j=1}^{k}\lambda_j = 1/k$. The Eq. (17) follows from the convention $0\log 0 = 0$ [21]. □

Assumption 1. *We assume that representation $\boldsymbol{h}$ follows zero-mean multivariate Gaussian distribution. In addition, we assume that the components of $\boldsymbol{h}$ (denoted as $\boldsymbol{h}^{(i)}$) have homogeneous variance of $\frac{1}{d}$, i.e., $\forall_{i=1}^{d}\boldsymbol{h}^{(i)} \sim \mathcal{N}(0, \frac{1}{d})$.*

Theorem 2 (Disentanglement and VNE). *Under the Assumption 1, $\boldsymbol{h}$ is disentangled if $S(\mathcal{C}_{auto})$ is maximized.*

Proof. By Assumption 1, $\boldsymbol{h} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_1)$ for $\boldsymbol{\Sigma}_1 \in \mathbb{R}^{d\times d}$ where diagonal entries in $\boldsymbol{\Sigma}_1$ are equal to $1/d$.

In addition, we define new random variable $\boldsymbol{h}' \sim \mathcal{N}(0, \boldsymbol{\Sigma}_2)$ for $\boldsymbol{\Sigma}_2 = \frac{1}{d}\cdot I_d$.

Then, because $\boldsymbol{h}^{(i)} \sim \mathcal{N}(0, \frac{1}{d})$ and $\boldsymbol{h}'^{(i)} \sim \mathcal{N}(0, \frac{1}{d})$ and the components of $\boldsymbol{h}'$ are independent,

$$\prod_{i=1}^{d} p(\boldsymbol{h}^{(i)}) = \prod_{i=1}^{d} p(\boldsymbol{h}'^{(i)}) = p(\boldsymbol{h}'). \tag{19}$$

By Lemma 1, $S(\mathcal{C}_{\text{auto}})$ is maximized if and only if

$$\forall_{j=1}^{d}\lambda_j = \frac{1}{d}, \tag{20}$$

where λ_j are eigenvalues of $\boldsymbol{\Sigma}_1 (= \mathbb{E}[\boldsymbol{h}\boldsymbol{h}^T] = \mathcal{C}_{\text{auto}})$.

Starting from Definition of total correlation $TC(\boldsymbol{h})$ in [1], we have

$$\begin{aligned}
2\cdot TC(\boldsymbol{h}) &= 2\cdot D_{\text{KL}}(p(\boldsymbol{h})\|\prod_{i=1}^{d} p(\boldsymbol{h}^{(i)})) && (21)\\
&= 2\cdot D_{\text{KL}}(p(\boldsymbol{h})\|p(\boldsymbol{h}')) && (22)\\
&= \text{tr}(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1) - d + \log\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|} && (23)\\
&= d - d + \log\frac{(1/d)^d}{(1/d)^d} = 0, && (24)
\end{aligned}$$

where Eq. (22) follows from Eq. (19), Eq. (23) follows from Lemma 2, and Eq. (24) follows from Eq. (20).

If $TC(\boldsymbol{h}) = 0$, the components of $\boldsymbol{h}$ are independent, therefore $\boldsymbol{h}$ is disentangled [1]. □

Theorem 3 (Isotropy and VNE). *For a given representation matrix $\boldsymbol{H} \in \mathbb{R}^{N\times d}$, suppose that $N \leq d$ and $S(\mathcal{C}_{auto})$ is maximized. Then,*

$$\boldsymbol{H}\boldsymbol{H}^T = \boldsymbol{I}_N. \tag{25}$$

Proof. We consider singular value decomposition of $\boldsymbol{H} (= \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T)$ for $\boldsymbol{U} \in \mathbb{R}^{N\times N}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times d}$, and $\boldsymbol{V} \in \mathbb{R}^{d\times d}$. If $N \leq d$ and $S(\mathcal{C}_{\text{auto}})$ is maximized, by Lemma 1, eigenvalues of $\mathcal{C}_{\text{auto}} (= \boldsymbol{H}^T\boldsymbol{H}/N = \boldsymbol{V}\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}\boldsymbol{V}^T/N)$ are supposed to be equal to $1/N$ for the first N eigenvalues and zero for the others. Therefore $\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T = \boldsymbol{I}_N$ and we have

$$\boldsymbol{H}\boldsymbol{H}^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T\boldsymbol{U}^T = \boldsymbol{I}_N. \tag{26}$$

□

C. Main Algorithm

```
# N   : batch size
# d   : embedding dimension
# H   : embeddings, Tensor, shape=[N, d]

def get_vne(H):
    Z = torch.nn.functional.normalize(H, dim=1)
    rho = torch.matmul(Z.T, Z) / Z.shape[0]
    eig_val = torch.linalg.eigh(rho)[0][-Z.shape[0]:]
    return - (eig_val * torch.log(eig_val)).nansum()

# the following is equivalent and faster when N < d
def get_vne(H):
    Z = torch.nn.functional.normalize(H, dim=1)
    sing_val = torch.svd(Z / np.sqrt(Z.shape[0]))[1]
    eig_val = sing_val ** 2
    return - (eig_val * torch.log(eig_val)).nansum()
```

Figure 9. PyTorch implementation of VNE.

D. Computational Overhead

We train I-VNE$^+$ using 2×RTX 3090 GPUs, ImageNet-1K, and various batch sizes and models. In Table 9, the average computational overhead is 2.68%.

Model		ResNet-18			ResNet-50		
Batch Size		256	128	64	256	128	64
Average training time per iteration (sec.)	On VNE	0.051	0.024	0.011	0.120	0.073	0.031
	Total	2.318	1.288	0.845	2.745	2.101	1.127
Overhead		2.21%	1.89%	1.36%	4.37%	3.48%	2.75%

Table 9. Computational overhead of VNE.

E. Experimental Details for I-VNE$^+$

The PyTorch implementation codes will be made available online. Our implementations follow the standard training protocols of SSL in [35, 96] and the standard evaluation protocols of SSL in [34, 35, 63, 96]. A few important hyperparameters are described as follows.

Backbone and Projector: For all datasets, we use ResNet-50 [38] as the default backbone. For CIFAR-10, we use 2-layer MLP projector with hidden dimension of 2048 and output dimension of 128. For ImageNet-100, we use 3-layer MLP projector with hidden dimension of 2048 and output dimension of 256. For ImageNet-1K, we use the same projector as in the ImageNet-100 case, except that the output dimension is 512.

Optimization: We use SGD optimizer with momentum of 0.9. The learning rate (LR) is linearly scaled with batch size (LR = base learning rate $\times$ batch size / 256), and it is scheduled by the cosine learning rate decay with 10-epoch warm-up [59]. For CIFAR-10 and ImageNet-100, we use base learning rate of 0.4, batch size of 64, and weight decay of 1e-4. For ImageNet-1K, we use base learning rate of 0.2, batch size of 512, and weight decay of 1e-5.

Augmentation: For CIFAR-10 and ImageNet-100, we adopt multi-view setting in [11] and generate 6 views using the same augmentations in [14] (for CIFAR-10) and in [11] (for ImageNet-100). For ImageNet-1K, we generate the default 2 views using the same augmentation as in [35]. Note that we use 2-view setting for ImageNet-1K because of the computational limitation.

F. Supplementary Results

Method	Top-1	Top-5
Supervised [14]	76.5	93.7
SimCLR [14]	69.3	89.0
MoCo v2 [17]	71.1	90.1
InfoMin Aug. [81]	73.0	91.1
BYOL [35]	74.3	**91.6**
SwAV [11]	**75.3**	
Shuffled-DBN [43]	65.2	
Barlow Twins [96]	73.2	91.0
VICReg [7]	73.2	91.1
I-VNE$^+$ (ours)	72.1	91.0

Table 10. SSL: Linear evaluation performance in ImageNet-1K for various representation learning methods. They are all based on ResNet-50 encoders pre-trained with various datasets. Linear classifier on top of the frozen pre-trained model is trained with labels. State-of-the-art methods are included and the best results are indicated in bold.

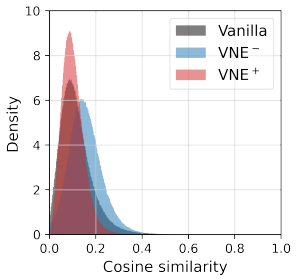

Figure 10. Meta-learning: Disentanglement of representation.

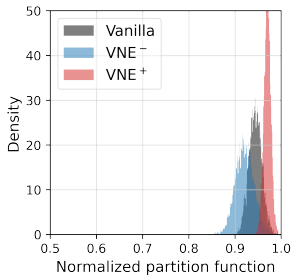

Figure 11. Domain generalization: Isotropy of representation.

Algorithm	Method	PACS		VLSC		OfficeHome		TerraIncognita	
		Avg.	Diff.	Avg.	Diff.	Avg.	Diff.	Avg.	Diff.
ERM	Vanilla	85.2		76.7		64.9		45.4	
	VNE$^-$	86.9	**1.7**	78.1	**1.4**	65.9	**1.0**	50.6	**5.2**
	SE$^-$	85.0	-0.2	76.5	-0.2	65.3	0.4	50.4	5.0
SWAD	Vanilla	88.2		79.4		70.2		50.9	
	VNE$^-$	88.3	0.1	79.7	**0.3**	71.1	**0.9**	51.7	**0.8**
	SE$^-$	88.4	**0.2**	79.6	0.1	71.0	0.8	51.2	0.2

Table 11. Von Neumann entropy vs. Shannon entropy: The results of domain generalization with ERM and SWAD algorithms are shown. For regularizing Shannon entropy, we have used the InfoNCE estimation of self-information, $I_{\mathrm{NCE}}(\boldsymbol{h};\boldsymbol{h})$.