DEEP HIERARCHICAL-HYPERSPHERICAL LEARNING (DH^{2}L)

Anonymous authors
Paper under double-blind review

ABSTRACT

Regularization is known to be an inexpensive and reasonable solution to alleviate over-fitting problems of inference models, including deep neural networks. In this paper, we propose a hierarchical regularization which preserves the semantic structure of a sample distribution. At the same time, this regularization promotes diversity by imposing distance between parameter vectors enlarged within semantic structures. To generate evenly distributed parameters, we constrain them to lie on *hierarchical hyperspheres*. Evenly distributed parameters are considered to be less redundant. To define hierarchical parameter space, we propose to reformulate the topology space with multiple hypersphere space. On each hypersphere space, the projection parameter is defined by two individual parameters. Since maximizing groupwise pairwise distance between points on hypersphere is nontrivial (generalized Thomson problem), we propose a new discrete metric integrated with continuous angle metric. Extensive experiments on publicly available datasets (CIFAR-10, CIFAR-100, CUB200-2011, and Stanford Cars), our proposed method shows improved generalization performance, especially when the number of super-classes is larger.

1 INTRODUCTION

Diversity promoting learning has been widely adopted via enlarging pairwise distances (Xie et al., 2018; 2017a; Liu et al., 2018), increasing orthogonality (Xie et al., 2018), reducing covariance between parameters (Xie et al., 2017b), or reducing correlation on feature (Cogswell et al., 2016) to improve generalization performance. Among them, *diversity promoting regularization* (Xie et al., 2017a) (Xie et al., 2017b) by enforcing large diversity between projection parameters achieves a reasonable performance without modifying the model structure. Optimizing the objective function with a covariance matrix in these methods is nontrivial. The diversity promoting regularization via minimizing energy of parameters of deep neural networks has been proposed (Liu et al., 2018). By minimizing a pairwise distance between parameters on hypersphere with the known metrics, they achieved the improved generalization performance.

Following an efficient regularization on hypersphere space, we explore further this direction with three main concepts (hierarchical and hyperspherical learning with discrete metrics).
1) *Why hierarchical learning?* Hierarchical inference explains *human intelligence*. In (Kurzweil, 2013), it states that "the neocortex contains about 300 million very general pattern recognizers, arranged in a hierarchy". Applying a hierarchy of multiple classes based on semantic taxonomy is a natural choice to devise *machine intelligence*. Effectiveness of the hierarchical learning can be found in (Verma et al., 2012).

2) *Why hyperspherical learning?* Hypersphere can be represented by a centroid and a radius. Due to the denominator in the unit-length normalization ($\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$), the distance defined on the hypersphere converges when the magnitude of $\boldsymbol{w}$ goes infinity while Euclidean distance goes infinity. Due to this bounded property, hierarchical structure with multiple separated hyperspheres can be defined.

3) *Why discrete metric learning?* If the vector points form discontinuous series with discrete representation (e.g. multi-dimensional binary or ternary), they are isolated from each other with a certain margin. This property may fit with disconnected manifold or groupwise space problem. Moreover, to make points to be equidistributed where a pairwise distance is maximized is a nontrivial task.

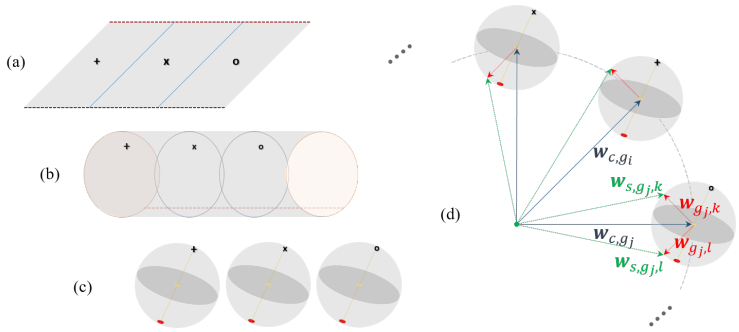


Figure 1: Multiple (hyper)spheres as quotient spaces of a topology space on Euclidean space might be found by gluing process with identifying points. Those separated hyperspheres are assumed to be under the quotient space conditions (Tu, 2010). Within an individual (hyper)sphere, the projection parameters in deep neural networks preserve a hierarchical structure. The space can be formed in a series: (a), (b), (c), and (d)

Because of finite an isolated points, this metric may reduce search efforts to satisfy those constraints using a set of pairwise distance.

In this paper, we propose to apply hierarchical structure to parameter regularization on the multiple groupwise hyperspherical spaces. In order to find an appropriate metric on this space, we explore a discrete angluar metric. We examine the proposed method on extensive experimental setups in terms of datasets and deep network models.

2 Multiple Separated Hyperspheres

Samples observed from the real world may be on disconnected manifold. In other words, disjoint union of those manifold could generate the global manifold (Lee, 2000). In this section, we decompose the one space into multiple spaces (manifolds) and re-define the space in terms of hierarchical point of view.

2.1 Disconnected Manifold via Equivalent Relations

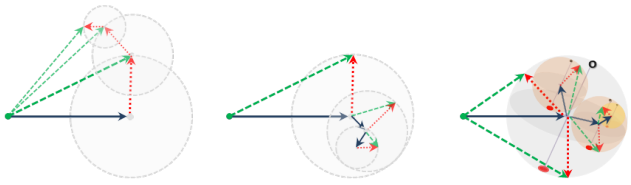
Since it is not suitable to measure a pairwise distance between high dimensional vectors which have the hierarchical structure in the same space, we construct another identification space which includes isolated spaces, from the original space (via equivalence relation (Tu, 2010)). Denote d-sphere $\mathbb{S}^d$ to be the set of points that satisfies $\mathbb{S}^d = \{\boldsymbol{w} \in \mathbb{R}^{d+1} : \|\boldsymbol{w}\| = 1\}$. We construct multiple separated hyperspheres using multiple identifying relations. In Figure 1, we use the center vector $\boldsymbol{w}_c$ and the surface vector $\boldsymbol{w}_s$ to define a hypershpere space and the projection parameter $\boldsymbol{w}$.

2.2 Prior Distribution and Regularization

To make the parameter vectors uniformly distributed on the unit hypersphere, the vectors are sampled from the Gaussian normal distribution (Muller, 1959; Harman & Lacko, 2010). This is because the normal distribution is spherically symmetric (Muller, 1959). In a Bayesian point of view, neural networks with Gaussian priors are known to induce an l^2-norm regularization (Vladimirova et al., 2019). From two evidences, we know that enforcing the parameters to have the Gaussian prior is important in hyperspherical learning in neural networks. Note that a parameter which is calculated from the difference arithmetic operation with two parameters on the normal Gaussian distribution is on the *normal difference distribution*.

3 Method

In deep neural networks, the objective function $\mathcal{J}$ with regularization $\mathcal{R}$ in addition to a loss $\mathcal{L}$, $\mathcal{J}_{\mathcal{R}(\mathbf{W})} = \mathcal{L}(\boldsymbol{x}, \mathbf{W}) + \lambda\mathcal{R}(\mathbf{W})$, is optimized to find the optimal $\mathbf{W}$ having a near minimum loss $\mathcal{L}$, $\arg\min_{\mathbf{W}} \mathcal{J}_{\mathcal{R}(\boldsymbol{x},\mathbf{W})}$, where $\boldsymbol{x} \in \mathbb{R}^{d_0}$ denotes an input vector, $\mathbf{W} = \{\boldsymbol{W}_i \in \mathbb{R}^{d_{i-1} \times c_i} : \boldsymbol{W}_i = \{\boldsymbol{w}_j \in$



(a) (b) (c)

Figure 2: (a) A radius of global area converges to $\frac{r_0}{1-\delta}$ ($= \sum_l^\infty r_0 \delta^l$: the sum of radius series, assuming δ: constant) as l goes to infinity where r_0 is their initial radius and the constant δ is the ratio between radiuses $\frac{r_l}{r_{l-1}}$ which the absolute value is less than one. (b) The radius of global area is bounded to the initial radius r_0 of a series of spheres. This bears a resemblance to the process of repeat of *Hypersphere packing* which arranges non-overlapping spheres within a containing space. (c) A bounded space is better to model. Following (b), hierarchical 2-sphere is defined and generalized to higher dimensional sphere, hypersphere ($\mathbb{S}^d$, $d \geq 3$).

$\mathbb{R}^{d_{i-1}}\}, j = 1, \ldots, c_i, i = 1, \ldots, L\}$ denotes a set of parameter matrices (i.e. neurons/kernels), L denotes the number of layers, and $\lambda > 0$ is to control the degree of the regularization. For a classification task, the cross entropy loss is used for the loss function $\mathcal{L}$. We propose a new regularization formulation $\mathcal{R}$ in Section 3.1.

3.1 Regularization for Hierarchical Hyperspherical Hypotheses

Denote $\boldsymbol{w}$ a projection parameter vector (an element of $\boldsymbol{W}$ at a single layer) to transform a given input into the embedding space defined in a Euclidean metric space: $\boldsymbol{x} \in \mathbb{R}^{d+1} \mapsto \boldsymbol{w}^T \boldsymbol{x} \in \mathbb{R}$. By the definition of unit-length projection $\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$, a new parameter $\hat{\boldsymbol{w}}$ can be defined on d-sphere: $\mathbb{S}^d = \{\hat{\boldsymbol{w}} \in \mathbb{R}^{d+1} : \|\hat{\boldsymbol{w}}\| = 1\}$ where $\|\cdot\|$ denotes l^2-norm and the center is zero. In other word, the projection parameter vector $\hat{\boldsymbol{w}}$ can be defined by a center point vector $\boldsymbol{w}_c \in \mathbb{R}^{d+1}$ and a surface vector $\boldsymbol{w}_s \in \mathbb{R}^{d+1}$ using an arithmetic operation: $\hat{\boldsymbol{w}} := \boldsymbol{w}_s - \boldsymbol{w}_c$. We define the d-sphere with the center and surface vector: $\mathbb{S}^d_{\boldsymbol{w}_c} = \{\boldsymbol{w}_s - \boldsymbol{w}_c \in \mathbb{R}^{d+1} : \|\boldsymbol{w}_s - \boldsymbol{w}_c\| = 1\}$. For a notation simplicity, we use $\boldsymbol{w}$ instead of $\hat{\boldsymbol{w}}$ hereafter. While we consider a radius equals to 1 for simplicity, the parameter vector can have a radius $r > 0$.

3.1.1 Hierarchical parameters derived from levelwise and groupwise centroid vectors

We assume that the hierarchical structure consists of levelwise structure with a notation (l) and groupwise structure with a notation g below. We explain these two concept to parameter vectors serially.

Levelwise structure The above parameter vectors on $\mathbb{S}^d_{\boldsymbol{w}_c}$ can be defined with the level-wise notation (l) as follows,

$$\boldsymbol{w}^{(l)} := \boldsymbol{w}_s^{(l)} - \boldsymbol{w}_c^{(l)} \tag{1}$$

where the parameters are defined on l-th d-sphere, $\mathbb{S}^d_{\boldsymbol{w}_c^{(l)}}$. In Figure 2, an example is provided in a lower dimension. In this paper, we define the hierarchical parameters in a higher dimensional space than that of (b) and (c) in Figure 2.

In a levelwise setting, $\boldsymbol{w}_s^{(l)}$ and $\boldsymbol{w}_c^{(l)}$ are additively represented based on the center parameter (centroid) calculated from the previous level: $\boldsymbol{w}_c^{(l-1)} + \overrightarrow{\Delta \boldsymbol{w}}^{(l)} \mapsto \boldsymbol{w}_c^{(l)}$, where $\boldsymbol{w}_c^{(l-1)} = \sum_i^{l-1} \overrightarrow{\Delta \boldsymbol{w}}^{(i)}$ is the accumulated center vector and $\overrightarrow{\Delta \boldsymbol{w}}^{(l)}$ denotes a newly connected parameter vector from $\boldsymbol{w}_c^{(l-1)}$ to $\boldsymbol{w}_c^{(l)}$. By denoting $\overrightarrow{\Delta \boldsymbol{w}}^{(l)}$ as $\boldsymbol{w}^{(l,l-1)}$, the center vector at the l-level is defined as, $\boldsymbol{w}_c^{(l)} := \boldsymbol{w}_c^{(l,l-1)} + \boldsymbol{w}_c^{(l-1)}$, and the surface vector $\boldsymbol{w}_s^{(l)} := \boldsymbol{w}_s^{(l,l-1)} + \boldsymbol{w}_c^{(l-1)}$. Both the center vector and the surface vector at the current level are based on the center vector at the previous level[1]. Hence, eq. (1) is equivalent to

$$\boldsymbol{w}^{(l)} = \boldsymbol{w}_s^{(l,l-1)} - \boldsymbol{w}_c^{(l,l-1)}. \tag{2}$$

[1]As not every sample has a child sample, it might be more reasonable to branch from representative parameter or center parameter rather than from individual projection parameters.

Note that we use $(l, l-1)$ to denote a connected parameter from the center parameter at the $(l-1)$th level to (l)th level.

Groupwise structure With a group notation g_k, Then the center parameter in eq. (1) can be rephrased as $\boldsymbol{w}_{c,g_k}^{(l,l-1)}$ on $\mathbb{S}^d_{\boldsymbol{w}_{c,g_k}^{(l,l-1)}}$, which is d-sphere of g_k group at l-th level, $g^{(l)} := \{g_k\}_{k=1}^{|g^{(l)}|}$, $g^{(l)} \subseteq \mathbb{G}^{(l)}$ is a group set at the lth level, and $|\cdot|$ denotes the cardinality. A group $g^{(l)}$ at the current level is conditioned on a group at the previous level $g^{(l-1)} := \{g_{k'}\}_{k'=1}^{|g^{(l-1)}|}$ where $g^{(l-1)} \subseteq \mathbb{G}^{(l-1)}$. With their groupwise relation over levels, an adjacency indication[2] $P^{(l,l-1)}(\{\mathbb{G}^{(l-1)}, \mathbb{G}^{(l)}\}) \in \{0,1\}^{|\mathbb{G}^{(l-1)}| \times |\mathbb{G}^{(l)}|}$ calculated. Hence, the parameter projection vector at the lth-level is determined as: $\boldsymbol{w}_{g_k,i}^{(l)} := \{\boldsymbol{w}_{s,g_k,i}^{(l,l-1)} - \boldsymbol{w}_{c,g_k}^{(l,l-1)}\}$ on $\mathbb{S}^d_{\boldsymbol{w}_c^{(l,l-1)},g_k}$ where $i = 1, \ldots, |g_k|$, $\{\boldsymbol{w}_{s,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}\}$ is calculated based on $\boldsymbol{w}_{c,g^{(l-1)}}^{(l-1)}$ referring to their group condition, and the adjacency matrix $P^{(l,l-1)}$.

A representative vector of the group g_k at (l) level is $\boldsymbol{w}_{c,g_k}^{(l)}$ which is equivalent to a mean vector of $\boldsymbol{w}_{s,g_k}^{(l)} \Rightarrow \mu(\boldsymbol{w}_{s,g_k}^{(l)}) = \frac{1}{|g_k|}\sum^{|g_k|} \boldsymbol{w}_{s,g_k}^{(l)}$. If the representative vector for the group g_k is determined by a certain parameter vector and the center vector at the previous level, then an adjust factor (ϵ) can be used: $\boldsymbol{w}_{c,g_k}^{(l,l-1)} = \boldsymbol{w}_{c,g_{k'}}^{(l-1)} + \epsilon \cdot \boldsymbol{w}_{g_{k'},i}^{(l-1)}$, where $\boldsymbol{w}_{g_{k'},i}^{(l-1)} \in \mathbb{S}^d_{\boldsymbol{w}_{c,g'_k}^{(l-1)}}$.

3.1.2 HIERARCHICAL REGULARIZATION

In this section, we define a regularization term of the hierarchical parameter vectors defined above. A set of parameters $\{\boldsymbol{W}_{s,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g'_k}^{(l-1)}\} \in \boldsymbol{W}\ \forall g_k,\ \forall g_{k'}$ where $\boldsymbol{W}_{s,g_k}^{(l,l-1)} := \{\boldsymbol{w}_{s,g_k,i}^{(l,l-1)}\}_{i=1}^{|g_k|}$, is an optimizing target of hierarchical regularization term as follows:

$$\mathcal{R}(\boldsymbol{W}) := \sum_l \lambda_l \mathcal{R}_l(\boldsymbol{W}_{s,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}; P^{(l,l-1)}) + \sum_l \mathcal{C}_l(\boldsymbol{w}_{c,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g'_k}^{(l-1)}; P^{(l,l-1)}) \quad (3)$$

where $\mathcal{R}_l$ works on individual spheres $\mathbb{S}^d_{\boldsymbol{w}_{c,g_k}^{(l,l-1)}}$, $\lambda_l \in \mathbb{R}_{>0}$, and $\mathcal{C}_l$ aims to apply geometry-aware constraints across spheres. $\mathcal{R}_l$ consists of two parts of regularization terms with: 1) $\mathcal{R}_{l,p}$ for projection parameter vectors in the same group g_k on $\mathbb{S}^d_{\boldsymbol{w}_{c,g_k}^{(l,l-1)}}$ and 2) $\mathcal{R}_{l,c}$ for center parameter vectors across the groups in the same level on $\mathbb{S}^d_{\boldsymbol{w}_{c,g'_k}^{(l-1)}}$,

$$\mathcal{R}_l(\boldsymbol{W}_{s,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}) := \mathcal{R}_{l,p}(\boldsymbol{W}_{s,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}) + \mathcal{R}_{l,c}(\boldsymbol{w}_{c,g_k}^{(l,l-1)}), \quad (4)$$

where

$$\mathcal{R}_{l,p}(\boldsymbol{W}_{s,g_k}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}) := \frac{1}{|g^{(l)}|}\frac{2}{G(G-1)} \sum_{\{g_k \in g^{(l)}\}} \sum_{\{i \neq j \in g_k\}} d(\boldsymbol{w}_{g_k,i}^{(l,l-1)}, \boldsymbol{w}_{g_k,j}^{(l,l-1)}), \quad (5)$$

$$\mathcal{R}_{l,c}(\boldsymbol{w}_{c,g_k}^{(l,l-1)}) := \frac{2}{C(C-1)} \sum_M d(\boldsymbol{w}_{c,g_i}^{(l,l-1)}, \boldsymbol{w}_{c,g_j}^{(l,l-1)}), \quad (6)$$

where $\boldsymbol{w}_{g_k,i}^{(l,l-1)} := \boldsymbol{w}_{s,g_k,i}^{(l,l-1)} - \boldsymbol{w}_{c,g_k}^{(l,l-1)}$, ($\boldsymbol{w}_{g_k,i}^{(l,l-1)}$ similarly defined), $G = |\{i \neq j \in g_k\}|$, $C = |\{g_i \neq g_j \in g^{(l)}\}|$, and $d(\cdot,\cdot)$ denotes a distance metric between the parameter vectors. The distance metric is defined in Section 3.2. When a (mini)batch of inputs is given, the regularization term becomes: $E(\mathcal{R}(\boldsymbol{W})) = \frac{1}{|m_x|}\sum_{m_x} \mathcal{R}(\boldsymbol{W}; m_x)$. We explain these parameter vectors and pairwise distances in Figure 4 in Appendix.

In addition to the above hierarchical regularization in eq. (3), the orthogonality promoting term can be applied to the center vector $\boldsymbol{w}_{c,g_k}^{(l,l-1)}$: $\arg\min_{\boldsymbol{W}_c^{(l,l-1)}} \lambda_o \|\boldsymbol{W}_c^{(l,l-1)^T} \boldsymbol{W}_c^{(l,l-1)} - \boldsymbol{I}\|_F$ where $\boldsymbol{W}_c^{(l,l-1)} \in \mathbb{R}^{d \times |g_k|}$, $\|\cdot\|_F$ is the Frobenius norm and $\lambda_o > 0$. The parameters without the hierarchical information can adopt the magnitude (l^2-norm) minimization ($\arg\min_{\boldsymbol{w}} \lambda_f \sum_k \|\boldsymbol{w}_k\|$, where

[2]This can be replaced with a probability model.

$\boldsymbol{w}_k \in \boldsymbol{W}$ and $\lambda_f > 0$.) and energy (pairwise distance) minimization ($\arg\min_{\boldsymbol{w}} \sum_{i \neq j} \lambda_e d(\boldsymbol{w}_i, \boldsymbol{w}_j)$, where $\lambda_e > 0$.).

The constraint term help construct geometry-aware relational parameters between different spheres on the same level and on the across levels. Multiple constraints are defined as $\mathcal{C}_l := \sum_k \lambda_k \mathcal{C}_{l,k}$, where $\mathcal{C}_{l,k}$ is kth constraint between parameters at lth and $(l-1)$th, and $\lambda_k > 0$ is a Lagrange multiplier. We apply three constraints in a geometric point of view. The detailed formulation is defined in appendix.

3.2 Discrete and continuous angular distance metric

Discrete (code) product metric might be a good fit with the above group-wise definition. We expect that a projected point from the parameters formed in a discrete metric space, are isolated from each other. In Figure 3, discrete distance helps a distribution of pairs having the same angle distance to be diversified. In order to maximize the distance between the parameters, maximization of discrete distance could help the distribution of parameters diverse.

Using parameter vectors $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ on $\mathbb{R}^{d+1}$, we define a discrete distance metric using a sign function as follows:

$$D_h := \frac{1}{d} \sum_{k}^{d} sign(\boldsymbol{w}_i(k)) \cdot sign(\boldsymbol{w}_j(k)), \tag{7}$$

where $sign(x) := \begin{cases} 1, & \text{if } x \geq 0 \\ -1, & \text{otherwise} \end{cases}$, $-1 \leq D_h \leq 1$, and $\boldsymbol{w} = \{\boldsymbol{w}(k) \mid \forall k = 1, \ldots, d\} \in \mathbb{R}^{d+1}\}$. This is a normalized version of Hamming distance. For a ternary discrete, $\{-1, 0, 1\}$ is used. In order to consider the discrete distance as an angle distance within $[0, 1]$, normalized one is defined as $D_{h01} := \frac{-D_h+1}{2}, 0 \leq D_{h01} \leq 1$. The angle distance based on the above product can be rephrased as $\theta_{D_h} = {D_{h01}}^3$ where $0 \leq \theta_{D_h} \leq 1$.

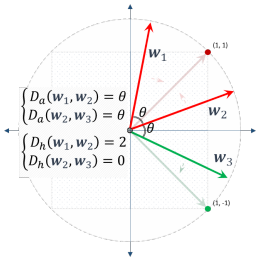


Figure 3: While the pairwise angle distances D_a between a pair of vectors $\{\boldsymbol{w}_1, \boldsymbol{w}_2\}$ and $\{\boldsymbol{w}_2, \boldsymbol{w}_3\}$ are the same, the pairwise discrete product distances D_h between vectors are different. To diversify a parameter space, the space with sign could be effective to recognize their difference.

As the discrete distance could be limited to approximate the model distribution. We merge the above discrete distance metric with continuous angle distance metric ($\theta = \frac{1}{\pi} \arccos\left(\frac{\boldsymbol{w}_i \cdot \boldsymbol{w}_j}{\|\boldsymbol{w}_i\| \|\boldsymbol{w}_j\|}\right), 0 \leq \theta \leq 1$) into the single metric. We simply use the definition of Pythagorean means which consist of the arithmetic mean (AM), the geometric mean (GM), and the harmonic mean (HM). Pythagorean means using the above angle pair is defined as follows:

$$D_{\text{AM}} := \frac{\theta_{D_h}+\theta}{2}, \quad D_{\text{GM}} := \theta_{D_h}\theta, \quad D_{\text{HM}} := \frac{4\theta_{D_h}\theta}{\theta_{D_h}+\theta} \tag{8}$$

In the angular distance[4] using $\{\theta_{D_h}, \theta\}$, a reversed form $1 - D_{\{\theta_{D_h}, \theta\}}$ is adopted to maximize an angle in optimization formulation as a form of minimization instead of $(\cdot)^{-s}$ where $s = 1, 2, \ldots$ which is used in Thomson problem that utilizes s-energy (Brauchart & Grabner, 2015).

The cosine similarity of these angles can be defined as follows:

$$D_{\cos(\text{AM})} := \cos\left(\frac{\theta_{D_h}+\theta}{2}\pi\right), D_{\cos(\text{GM})} := \cos\left(\theta_{D_h}\theta\pi\right), D_{\cos(\text{HM})} := \cos\left(\frac{4\theta_{D_h}\theta}{\theta_{D_h}+\theta}\pi\right), \tag{9}$$

then the cosine similarity functions are normalized with $\frac{\cos(\cdot)+1}{2}$ to have a distance value within $[0, 1]$.

Finally, Pythagorean means of each cosine similarity can be calculated as follows:

$$D_{\text{AM}_{\cos}} := \frac{\cos\theta_{D_h}\pi+\cos\theta\pi+2}{4}, D_{\text{GM}_{\cos}} := \frac{(\cos\theta_{D_h}\pi+1)(\cos\theta\pi+1)}{4}, D_{\text{HM}_{\cos}} := \frac{(\cos\theta_{D_h}\pi+1)(\cos\theta\pi+1)}{\cos\theta_{D_h}+\cos\theta+2}. \tag{10}$$

[3] On the other hand, the angle can be considered as a cosine similarity directly, $D_h := \cos\theta_{D_h}\pi$. So as to get the angle distance, it needs an arccosine function $\theta_{D_h} = \frac{1}{\pi}\arccos D_h$. In summary, for the angle distance θ_{D_h}, either "D_{h01}" or "$D'_{h01} = \frac{1}{\pi}\arccos D_h$" where $0 \leq D_{h01} \leq 1$, can be adopted.

[4] In $0 \leq \theta \leq 1$, the angle and its cosine value show an inverse relationship:$0 \leq \theta \leq 1 \rightarrow 1 \geq \cos\theta\pi \geq -1$.

The above metric functions defined in (8), (9), and (10) satisfy the metric conditions: non-negativity, symmetry, and triangle inequality. The distance using the above metric functions between any two points is bounded, because the hypersphere is a compact manifold.

3.3 GRADIENTS AND BACKPROPAGATION

As the sign function is not differentiable at the value 0, we adopt alternative backpropagation function. We adopt straight-through estimator (STE) (Bengio et al., 2013) in the backward path of the neural networks for the sign function in the discrete metric. The derivative of the sign function is substituted with $1_{|w|\leq 1}$ in the backward pass, known as the saturated STE. As the derivative of $\arccos(x)$ ($\frac{-1}{\sqrt{1-x^2}}$) is undefined at the value $x = \pm 1$, we apply clamping to the cosine function to have $x \in [-0.99, 0.99]$[5] where $x = \cos(\theta\pi), 0 \leq \theta \leq 1$.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets We conduct the experiments using four publicly available datasets including small size images (CIFAR-10 and CIFAR-100) and large size images (CUB200-2011 (Wah et al., 2011) and Stanford-Cars (Krause et al., 2013b), shortly CUB200 and Cars respectively). CUB200 and Cars datasets are used for a fine-grained visual categorization. The fine-grained visual categorization is challenging due to their high intra-class variances and low inter-class variances. CIFAR-10 dataset is used to validate effectiveness of the proposed metric. Except CIFAR-10, we use two-level hierarchy pairs $\{parent, child\}$. In Table 7 at Appendix, statics of datasets in detail is provided.

Deep neural network models and training setting We adopt different size networks along the datasets. We adopt the deep residual network (*resnet*) (He et al., 2016) with smaller amount of parameters (light models, resnet-20 (0.29M) and resenet-110 (1.73 M)) for a small size input (32×32 pixels) such in CIFAR-10 and CIFAR-100 so as not to have redundant parameters leading to overfitting. The original resnet with larger amount of parameters (heavy models, Resnet-18 (11.28M) and Resnet-50 (23.91M)) which is available from the pytorch library for a fine-grained input (224×224 pixels) for CUB200 and Cars.

We applied hieararchical regularization in the FC layer. Mini-batches, 512 for light models and 256 for heavy models, are used in the SGD optimizer. In training with the hierarchical regularization, we assume that the global hierarchical structure is not given. Instead, stochastic or partial hierarchical structure is given within the given mini-batch and the label pairs. Even though SGD is known as an unbiased estimation, stochastic hierarchical pairs could affect the overall approximation performance upon a size of class pairs. Settings in more detail are provided in Appendix.

4.2 RESULTS

Object classification The method with pairwise distance based ('E'nergy) regularization ('E' in Table 1) performs better than the baseline as shown in Table 1. The discrete angular metric based regularization (D_h^{ter} (ternary code), D_h^{bin} (binary code), $D_{\cos\,(\mathrm{HM})}$, and $D'_{\mathrm{HM}_{\cos}}$) can improve the generalization performance in term of test accuracy compared to the other metrics such as angular2 ($\sum_{i\neq j} \arccos\,(\frac{\boldsymbol{w}_i\cdot\boldsymbol{w}_j}{\|\boldsymbol{w}_i\|\|\boldsymbol{w}_j\|})^{-2}$), cosine ($\sum_{i\neq j} \frac{\boldsymbol{w}_i\cdot\boldsymbol{w}_j}{\|\boldsymbol{w}_i\|\|\boldsymbol{w}_j\|}$), and N-euclidean2 ($\sum_{i\neq j} \|\frac{\boldsymbol{w}_i}{\|\boldsymbol{w}_i\|} - \frac{\boldsymbol{w}_i}{\|\boldsymbol{w}_i\|}\|^{-2}$)[6] where a normalized version of Euclidean. '2' is from Riesz s-energy and is set where higher accuracy is shown. Due to the unit-length projection for Euclidean (N-euclidean2), their performance is comparable to that other angular metrics. D' denotes a distance used D'_h from footnote 3 in Section 3.2. The regularization terms are applied over convolutional layers and FC layers. As l^2-norm minimization based regularization shows much improvement, we use l^2 regularization by default for experiments. As there are many metrics proposed, in the table, more meaningful metrics are shown.

[5] $x = \{0.99 \cdot 1_{x>0.99}, x, -0.99 \cdot 1_{x<-0.99}\}$

[6] $= \sum_{i\neq j} (2 - 2\frac{\boldsymbol{w}_i\cdot\boldsymbol{w}_j}{\|\boldsymbol{w}_i\|\|\boldsymbol{w}_j\|})^{-1}$

Table 1: CIFAR-10, Test Accuracy (%), resnet-20, E: Energy (pairwise distance) minimization, l^2:l^2-norm minimization

metric	E	E+l^2
baseline	90.34	92.21
N-euclidean2	90.93	92.35
angular2	90.47	92.38
cosine	90.53	92.40
D_h^{ter}	90.67	92.48
D_h^{bin}	90.67	92.48
$D_{\cos(\text{HM})}$	90.84	92.93
$D'_{\text{HM}_{\cos}}$	90.94	92.69

Table 2: CIFAR-100, Test accuracy (%).

	resnet-20		resnet-110	
metric	E	E+H	E	E+H
baseline	63.86	-	62.02	-
baseline$_{l^2}$	68.03	-	72.90	-
N-euclidean2	67.59	68.65	73.95	73.96
angular2	67.83	67.76	74.40	73.89
cosine	68.11	68.45	73.37	73.37
D_h^{ter}	68.44	68.68	73.73	73.97
D_h^{bin}	68.52	68.69	73.97	74.26
D_{AM}	68.58	68.86	73.43	73.50
$D_{\cos(\text{AM})}$	68.58	68.60	73.14	73.65
$D'_{\cos(\text{AM})}$	67.57	68.36	73.14	73.72
$D'_{\cos(\text{HM})}$	68.62	68.65	73.07	73.65

As shown in Table 2, the regularization shows significantly better performance than that of the baseline (without regularization) for both resnet-20 and resnet-100 on CIFAR-100 dataset. Comparing to the baseline$_{l^2}$, pairwise distance based 'E' regularization (D_h^{ter}, $D_{\cos(\text{AM})}$, $D'_{\cos(\text{HM})}$) shows better performance than other metrics. If the hierarchical 'H' regularization is applied, the generalization is improved further in most cases of both resnet-20 and resnet-110. As the binary metric shows a better performance than that ternary, we adopt binary discretization for the proposed discrete angular metrics ($D_\bullet$, $D_{\cos(\bullet)}$, $D_{\bullet_{\cos}}$) in the experiments.

Ablation study We experiment how the metrics affect the generalization performance. As shown in Table 3, the proposed method shows significantly improved performance compared to the baseline (l^2). Individual averaging settings (AM, GM, and HM) show different improvement patterns.

We examine to apply different metrics between convolutional layers (pairwise energy 'E' regularization) and fully connected layers (with hierarchical 'H' regularization) using resnet-20 and CIFAR-100 datasets. As shown in Table 4, the cases applying hierarchical regularization show better performance than the baseline applying only pairwise distance based 'E' regularization. In this experiment, a combination GM and HM shows a better improvement than that other combinations.

Fine-grained visual categorization In this experiment, we use two Fine-grained category datasets. One is about the birds (CUB200) and another is about the cars (Cars) which focus on single species of object. Based on species of Birds, pairs of $\{parent, child\}$ are generated per sample by the academically from the expert. Birds are based on variety of characteristics, whereas the cars are categorized by manually based on model names by non-expert. The rate between the number of superclass (parent) per subclass of CUB200 (0.35) is much larger than that of Cars (0.0459) (as shown in Table 7 at Appendix). That rate of Cars is smaller than that of CIFAR-100.

As shown in Table 5, the proposed hierarchical regularization significantly improve the test accuracy along the all metrics for both Resnet-18 and Resnet-50. Compared to the CUB200, as shown in Table 6, the improvement of the proposed method is not that significant in Cars dataset. This might be because CUB200 dataset has more the hierarchical categorization cases of superclasses and subclasses pairs.

Table 3: CIFAR-100, Test accuracy (%), resnet-20

metric	E+H
baseline$_{l^2}$	68.03
D'_{AM}	68.64
D'_{GM}	68.70
D'_{HM}	68.80
$D_{\text{AM}_{\cos}}$	69.24
$D_{\text{GM}_{\cos}}$	68.55
$D_{\text{HM}_{\cos}}$	68.77
$D'_{\text{AM}_{\cos}}$	68.96
$D'_{\text{GM}_{\cos}}$	69.00
$D'_{\text{GM}_{\cos}}$	68.83

Table 4: CIFAR-100, Test accuracy (%), heterogeneous metrics on (Conv. and FC), resnet-20

metrics (in conv., in FC)	E+H
baseline (l^2, l^2)	68.03
baseline (D_{GM}, l^2)	68.22
(D_{GM}, D_{AM})	68.58
(D_{GM}, D_{GM})	68.62
(D_{GM}, D_{HM})	69.04
(D_{GM}, D_{AM})	68.62
(D_{GM}, D_{GM})	68.65
(D_{GM}, D_{HM})	68.70

Table 5: CUB200, Test accuracy (%)

	Resnet-18		Resnet-50	
metric	E	E+H	E	E+H
baseline	72.17	-	74.21	-
baseline$_{l^2}$	72.29	-	74.05	-
N-euclidean2	72.61	75.99	73.49	76.14
angular2	72.43	76.11	73.55	76.66
cosine	72.12	75.64	73.26	76.85
D_h^{ter}	72.58	75.99	73.57	76.37
D_h^{bin}	72.55	76.21	73.57	76.99
D_{AM}	73.04	76.02	73.88	75.95
$D_{\text{AM}_{\cos}}$	72.31	76.14	73.59	77.32
$D'_{\text{AM}_{\cos}}$	72.28	75.37	72.42	74.12
D_{GM}	72.90	76.35	74.16	75.30
$D'_{\text{HM}_{\cos}}$	72.55	76.11	74.64	76.94
$D'_{\cos(\text{HM})}$	72.55	76.32	72.86	76.56

Table 6: Cars, Test accuracy (%)

	Resnet-18		Resnet-50	
metric	E	E+H	E	E+H
baseline	85.10	-	87.99	-
baseline$_{l^2}$	85.58	-	87.92	-
N-euclidean2	85.48	85.56	87.96	87.97
angular2	85.11	85.13	88.34	87.85
cosine	85.57	85.73	88.01	87.86
D_h^{ter}	85.35	85.99	85.35	88.07
D_h^{bin}	86.22	85.99	88.32	88.14
D_{AM}	85.66	85.66	88.39	88.11
$D_{\text{AM}_{\cos}}$	85.52	86.05	87.92	88.57
$D'_{\text{AM}_{\cos}}$	85.76	86.43	88.07	87.96
$D'_{\cos(\text{AM})}$	85.54	85.52	88.22	88.13

5 Related works

Promoting of diversity on embedding space or model parameters is widely adopted concept in machine learning related area to improve the generalization performance (Cogswell et al., 2016), (Yang et al., 2019), (Li et al., 2012), (Ratzlaff & Fuxin, 2019), (Xie et al., 2017b), (Xie et al., 2018), (Xie et al., 2017a), (Liu et al., 2018). The diversity exists at a variety of levels such as in feature level (Cogswell et al., 2016; Xie et al., 2018), in projection parameter level (Xie et al., 2017a; Liu et al., 2018), in model ensemble level (Zhou et al., 2018; Ratzlaff & Fuxin, 2019), in latent space model level (Ratzlaff & Fuxin, 2019; Liu et al., 2018), or in generative model level (Yang et al., 2019; Ratzlaff & Fuxin, 2019). Throughout the existing work, in other point of views, the authors utilized enlarging pairwise distance between features or parameters (Xie et al., 2018; 2017a; Liu et al., 2018), increasing orthogonality (Xie et al., 2018), reducing covariance between projection parameters (Xie et al., 2017b), or reducing correlation on feature (Cogswell et al., 2016).

Among the above approaches, enlarging pairwise distance between features requires computational efforts due to their covariance matrix. To optimize the solution, via singular value decomposition, unit-eigenvalue is utilized in (Xie et al., 2017b). From the non-convex optimization problem, another stabilization process such as convex relaxation (Xie et al., 2018) is utilized. To optimize a direction and magnitude of the parameter vector alternatively, they adopts an alternating direction method of multipliers (ADMM) (Xie et al., 2017a).

In terms of learning on hyperspherical space, (Liu et al., 2017) proposed that hyperspherical convolution (SphereCov) replaces the traditional inner-product based convolution in order to conduct learning angular representation on hyperspheres. By making magnitude of vectors during inner-product operation, learning could be more efficient and stable. To maximize distances between parameters, Minimum Hyperspherical Energy (Liu et al., 2018), is proposed to regularization methods to make parameters equidistributed globally.

6 Conclusion

We proposed the regularization method, which utilizes pairwise and groupwise relation between parameters. To define a hierarchical parameter space, we reformulated the topology space with multiple hypersphere space. On each hypersphere, projection parameter is determined by the surface parameter at the center parameter, which is constructed from that of the previous level. By imposing maximum pairwise angular distance between the projection parameter vectors, diversity of parameters preserving semantic structure is promoted. As the optimization process on hypersphere space is non-trivial, we proposed the discrete metric integrated with continuous metric. Extensive experiments using publicly available datasets (CIFAR-10, CIFAR-100, CUB200-2011, and Stanford Cars), the deep neural network with our proposed regularization showed superior classification performance, especially when the number of super-classes is larger.

REFERENCES

Y. Bengio, Nicholas Lonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv*, Aug. 2013.

Johann S. Brauchart and Peter J. Grabner. Distributing many points on spheres. *J. Complex.*, 31(3): 293–326, June 2015.

Michael Cogswell, Faruk Ahmed, Ross B. Girshick, Larry Zitnick, and Dhruv Batra. Reducing Overfitting in Deep Networks by Decorrelating Representations. In *International Conference on Learning Representations*, 2016.

Radoslav Harman and Vladimr Lacko. On decompositional algorithms for uniform sampling from n-spheres and n-balls. *Journal of Multivariate Analysis*, 101(10):2297 – 2304, 2010.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778, 2016.

Jonathan Krause, Jun Deng, Michael Stark, and Li Fei-Fei. Collecting a large-scale dataset of fine-grained cars. In *Second Workshop on Fine-Grained Visual Categorization (FGVC2)*, 2013a.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013b. Ddataset available at `http://imagenet.stanford.edu/internal/car196/`.

Ray Kurzweil. *How to Create a Mind: The Secret of Human Thought Revealed*. Penguin Books, New York, NY, USA, 2013.

J.M. Lee. *Introduction to Topological Manifolds*. Graduate texts in mathematics. Springer, 2000.

Nan Li, Yang Yu, and Zhi-Hua Zhou. Diversity regularized ensemble pruning. In *Proceedings of the 2012th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I*, ECMLPKDD'12, pp. 330–345, 2012.

Weiyang Liu, Yan-Ming Zhang, Xingguo Li, Zhiding Yu, Bo Dai, Tuo Zhao, and Le Song. Deep hyperspherical learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 3950–3960, 2017.

Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song. Learning towards minimum hyperspherical energy. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 6225–6236, USA, 2018. Curran Associates Inc.

Mervin E. Muller. A note on a method for generating points uniformly on n-dimensional spheres. *Commun. ACM*, 2(4):19–20, April 1959.

Neale Ratzlaff and Li Fuxin. HyperGAN: A generative model for diverse, performant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 5361–5369, 09–15 Jun 2019.

L.W. Tu. *An Introduction to Manifolds*. Universitext. Springer New York, 2010.

N. Verma, D. Mahajan, S. Sellamanickam, and V. Nair. Learning hierarchical similarity metrics. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2280–2287, June 2012.

Mariia Vladimirova, Jakob Verbeek, Pablo Mesejo, and Julyan Arbel. Understanding priors in Bayesian neural networks at the unit level. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6458–6467, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. Dataset available at `http://www.vision.caltech.edu/visipedia-data/CUB-200-2011/`.

Pengtao Xie, Yuntian Deng, Yi Zhou, Abhimanu Kumar, Yaoliang Yu, James Zou, and Eric P. Xing. Learning latent space models with angular constraints. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pp. 3799–3810, 2017a.

Pengtao Xie, Aarti Singh, and Eric P. Xing. Uncorrelation and evenness: A new diversity-promoting regularizer. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 3811–3820. JMLR.org, 2017b.

Pengtao Xie, Wei Wu, Yichen Zhu, and Eric P. Xing. Orthogonality-promoting distance metric learning: Convex relaxation and theoretical analysis. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 5399–5408, 2018.

Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tiangchen Zhao, and Honglak Lee. Diversity-sensitive conditional generative adversarial networks. In *International Conference on Learning Representations*, 2019.

Tianyi Zhou, Shengjie Wang, and Jeff A Bilmes. Diverse ensemble evolution: Curriculum data-model marriage. In *Advances in Neural Information Processing Systems 31*, pp. 5905–5916. Curran Associates, Inc., 2018.

APPENDIX

Dataset acquisition details CIFAR-100 dataset provides their labels. In CUB200 dataset, the pairs can be extracted from their filename. In Cars dataset, we parsed each fine label to one of nine coarse vehicle types, such as "Sedan", "SUV", "Van" and etc., following (Krause et al., 2013a).

Table 7: Statistics of benchmark datasets

Dataset	#classes $\{pa, ch\}$	#train	#test	input size	#samples /class	#super /subclass
CIFAR-10	$\{1, 10\}$	50,000	10,000	32×32	5000.00	0.1000
CIFAR-100	$\{20, 100\}$	50,000	10,000	32×32	500.00	0.2000
CUB200	$\{70, 200\}$	5,994	5,794	224×224	29.97	0.3500
Cars	$\{9, 196\}$	8,144	8,041	224×224	41.55	0.0459

Deep neural network models and training details First, resnet-20 (0.29M) and resenet-110 (1.73 M), which include a combination of Basic blocks with output channels $[16, 30, 64]$ are adopted for light models. An input dimensionality of the fully connected (FC) layer (a classier) is 64 for both resnet-20 and resenet-110. Second, heavy models are adopted such as Resnet-18 (11.28M[7]) and Resnet-50 (23.91M) which consists of the basic blocks (Resnet-18) or the bottelneck blocks with output channels [64, 128, 256, 512] in Conv. layers. An input dimensionality of the FC layer is 512 for Resnet-18 and 2048 for Resnet-50.

The networks are optimized with SGD for both light and heavy models: we fixed i) the weight initialization with Random-Seed number 0 in pytorch, ii) learning rate schedule [0.1, 0.01, 0.001], 3) with momentum 0.9, 4) regularization: l^2-norm minimization with $\lambda_f = 0.0005$, 5) orthogonalization $\lambda_o = 0.0001$, energy minimization $\lambda_e = \{0.1, 1\}$, and hierarchical minimization $\lambda_l = 0.1 \times \{1, 5\}$. All the regularization is not applied to the parameters of BatchNorm layers. A bias term in the FC layer is not used. The images in training and test, images are resized to 256 size. The images is

[7]Dependent on the number of classes and the corresponding center parameters, the size could variate (e.g. 11.42M for CUB200, 11.31M for Cars).

cropped with 224×224 size at random location in training and at center location in test. Horizontal flipping is applied in training. The light models are trained from scratch without the pretrained weights for 300 epochs. The heavy model is trained using pretrained model provided by pytorch library[8] with 100 epochs. The experiments are conduced using GPU "NVIDIA TESLA P40".

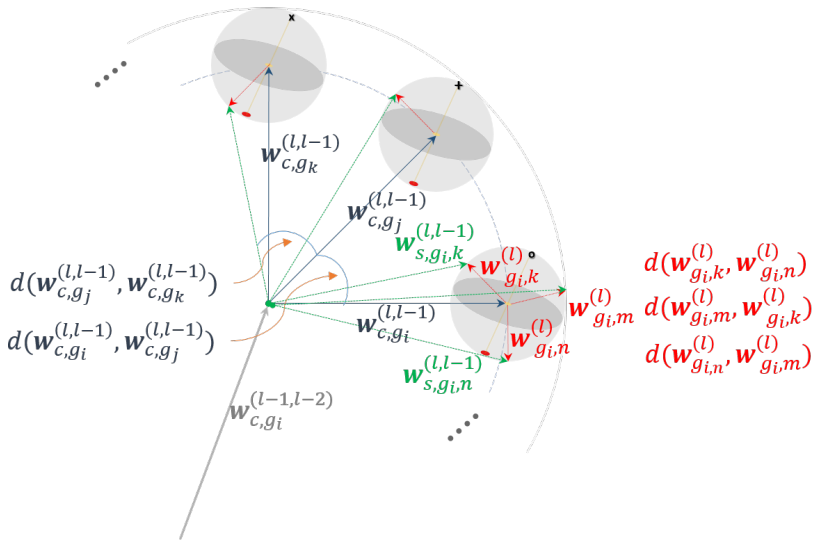


Figure 4: $\mathcal{R}_{l,p}(\boldsymbol{W}_{s,g_i}^{(l,l-1)}, \boldsymbol{w}_{c,g_i}^{(l,l-1)}) = d(\boldsymbol{w}_{g_i,n}^{(l,l-1)}, \boldsymbol{w}_{g_i,m}^{(l,l-1)}) + d(\boldsymbol{w}_{g_i,m}^{(l,l-1)}, \boldsymbol{w}_{g_i,k}^{(l,l-1)}) + d(\boldsymbol{w}_{g_i,k}^{(l,l-1)}, \boldsymbol{w}_{g_i,n}^{(l,l-1)}) + \dots$ corresponds to Eq (5) and $\mathcal{R}_{l,c}(\boldsymbol{w}_{c,g_k}^{(l,l-1)}) = d(\boldsymbol{w}_{c,g_i}^{(l,l-1)}, \boldsymbol{w}_{c,g_j}^{(l,l-1)}) + d(\boldsymbol{w}_{c,g_j}^{(l,l-1)}, \boldsymbol{w}_{c,g_k}^{(l,l-1)}) \dots$ corresponds to Eq (6)

***Constraints* in Eq. (3)** $\mathcal{C}_l := \sum_k \lambda_k \mathcal{C}_{l,k}$ can be given as follows:

1. *Constraint 1* ($\mathcal{C}_1$): This constraint describes that a radius of an inner sphere must be smaller than that of its outer sphere.
   $r^{(l-1)} - r^{(l)} \geq 0 \Rightarrow \|\boldsymbol{w}^{(l-1)} - \boldsymbol{w}^{(l)}\| = \|\boldsymbol{w}_s^{(l-1)} - \boldsymbol{w}_c^{(l-1)}\| - \|\boldsymbol{w}_s^{(l)} - \boldsymbol{w}_c^{(l)}\| \geq 0.$
2. *Constraint 2* ($\mathcal{C}_2$): This constraint describes that a center of an inner sphere must be located in its outer sphere.
   $r^{(l-1)} - (\|\boldsymbol{w}_c^{(l,l-1)}\| + r^{(l)}) \geq 0 \Rightarrow r^{(l-1)} - (\|\boldsymbol{w}_c^{(l-1,0)} - \boldsymbol{w}_c^{(l,0)}\| + r^{(l)}) = \|\boldsymbol{w}_s^{(l-1,0)} - \boldsymbol{w}_c^{(l-1)}\| - (\|\boldsymbol{w}_c^{(l-1)} - \boldsymbol{w}_c^{(l)}\| + \|\boldsymbol{w}_s^{(l)} - \boldsymbol{w}_c^{(l)}\|) \geq 0.$
3. *Constraint 3* ($\mathcal{C}_3$): This constraint describes that a margin between spheres must be larger than zero.
   $\|\boldsymbol{w}_c^{(l,l-1)}\|(2 - 2\cos\theta)^{0.5} - 2r^{(l)} \geq 0 \Rightarrow \|\boldsymbol{w}_c^{(l)}\|(2 - 2\frac{\sum_{i \neq j} \boldsymbol{w}_c^{(l),i} \cdot \boldsymbol{w}_c^{(l),j}}{\|\boldsymbol{w}_c^{(l)}\|^2})^{0.5} - 2\|\boldsymbol{w}_s^{(l)} - \boldsymbol{w}_c^{(l)}\|,$ where $\|\boldsymbol{w}_c^{(l,l-1)}\|(2 - 2\cos\theta)^{0.5} = \|\boldsymbol{w}_c^{(l,l-1)}\|(r^{(l-1)}\sin\theta^2 - (r^{(l-1)} - r^{(l-1)}\cos\theta)^2)^{0.5}.$

[8] from https://download.pytorch.org/models/resnet18-5c106cde.pth, https://download.pytorch.org/models/resnet50-19c8e357.pth respectively