O-VIT: ORTHOGONAL VISION TRANSFORMER

Anonymous authors
Paper under double-blind review

ABSTRACT

Inspired by the tremendous success of self-attention mechanism in natural language processing, the Vision Transformer (ViT) creatively applies it to image patch sequences and achieves incredible performance. However, ViT brings about feature redundancy and low utilization of model capacity. To address this problem, we propose a novel and effective method named Orthogonal Vision Transformer (O-ViT), to optimize ViT from the geometric perspective. O-ViT limits parameters of self-attention blocks to reside on the orthogonal manifold, which can reduce the similarity between trainable parameters and construct a higher degree of distinction between features. Moreover, O-ViT achieves both orthogonal constraints and negligible optimization overhead by adopting a surjective mapping between the orthogonal group and its Lie algebra. Comparative experiments on various image recognition tasks demonstrate the validity of O-ViT. The experimental results show that O-ViT can boost the performance of ViT by up to 6.4%.

1 INTRODUCTION

Recent years have witnessed Vision Transformer (ViT) taking over Convolution Neural Network (CNN) and achieving dramatic success in computer vision, such as image classification Touvron et al. (2021a); Yuan et al. (2021). ViT benefits from transferring the self-attention mechanism Vaswani et al. (2017) from language sequences to vision tasks to learn the internal characteristics of image patch sequences Kolesnikov et al. (2021). CNN kernels have a local view, which needs to be expanded layer by layer. By comparison, the self-attention mechanism allows ViT to obtain global features even in shallow layers Kolesnikov et al. (2021). Nonetheless, linear transformations (e.g., generations of query, key and value matrices) in the self-attention of ViT bring about *feature redundancy* and *exploding or vanishing outputs*, which restricts ViT to find the optimal solution or slower its optimization. On the one hand, the redundancy of learned features is related to the similarity between row or column vectors of trainable parameters. On the other hand, if the linear transformation expands or shrinks the norm of input vectors, the final output of ViT may be too "inflated" or "degraded", since there is a nest of full connection layers on top of self-attention blocks.

This motivates us to explore the optimization of ViT on the orthogonal manifold, where parameters are less redundant and can learn more discriminative features Wu et al. (2021b). To achieve this goal, we put forward a novel method named Orthogonal Vision Transformer (O-ViT). Each Matrix A that resides on the orthogonal manifold satisfies $A^TA = AA^T = E$ Wang et al. (2020), where E is an identity matrix. On the one hand, the orthogonal coefficient matrix of linear transformations has mutually orthogonal row or column vectors, which have low similarity and can learn essential features, further alleviating over fitting and improving generalization ability. There has been work to use orthogonality to realize dimension reduction and feature selection Wu et al. (2021b). On the other hand, given that $\|AX\| = (AX)^T(AX) = X^TA^TAX = \|X\|$, orthogonal transformations can maintain the length of input vector, mitigating exploding or vanishing final outputs. Compared with layer normalization, it is more intuitive to keep vector norm unchanged at early stages of training from a geometric view. The norm keeping property can also alleviate explosion and vanishing gradients in the rectified linear unit (ReLU) function Arjovsky et al. (2016). Furthermore, rounding error will accumulate in the forward calculation of ViT, since parameters are stored in decimal form. On account of eigenvalues with length $\|\cdot\|$ of absolute 1, orthogonal transformations are insensitive to rounding error and have numerical stability Lahlou & Oppenheim (2016). Moreover, orthogonal manifold reduces the dimension of solution space, leading to a faster convergence speed.

Optimizing on the manifold Smith (1994) has achieved impressive performance in deep learning Wang et al. (2020); Arjovsky et al. (2016). For instance, Huang & Gool (2017); Huang et al. (2018) utilize geometry constraints to construct analogous-convolution architecture. Moreover, orthogonal parameterization is proved to reduce filter similarities, preserve energy Wang et al. (2020), make spectra uniform Zhou et al. (2008), and stabilize the activation distribution in different network layers Rodríguez et al. (2017). Furthermore, orthogonal initializations of parameters can yield depth-independent learning times Saxe et al. (2014). However, there is few work to conduct orthogonal optimization in ViT, and this paper aims to bridge the gap between ViT and geometry optimization.

The gradient backpropagation is difficult in geometry optimization Smith (1994), since updating trainable parameters along the manifold involves extensive orthogonal projection and retraction operation calculation Bronstein et al. (2021). We pay attention to a surjective mapping between the orthogonal group and its Lie algebra, allowing O-ViT to transform computation-expensive geometry optimization into a general optimization problem in Euclidean space. Another way to achieve cheap optimization is to substitute a hard orthogonal regularizer for optimizing on the manifold, which has been widely used in orthogonal CNN and RNN Wang et al. (2020). They use $\|A^TA - E\|_F^2$ as a penalty term of the main task. Even an earlier practise of orthogonal ViT used the above orthogonal regularizers Zhang et al. (2021). Our approach does not adopt hard or soft orthogonal regularizers ($\lambda\|A^TA - E\|_F^2, \lambda$ is a hyperparameter) due to the following issues: i) training cost of imposing penalty term for a trade-off is high; ii) a trade-off may fail to converge to an optimal point that satisfies both main task and orthogonal regularization; and iii) the result of soft orthogonal regularizers partly depends on the hyperparameter λ, which is unreliable. Compared with hard or soft-orthogonality, our approach based on direct parameterization is clearer and more concise.

This paper makes the following three major contributions:

1. We propose a novel method named O-ViT to restrict space-projection parameters in self-attention to be on the orthogonal manifold and aggregate multiple orthogonal self-attention blocks, which is the first to improve ViT in a geometric optimization way.
2. O-ViT can pull the geometric optimization back to Euclidean optimization. As a result, O-ViT can be optimized by general gradient descent optimizers, which avoids complex orthogonal projection and retraction. Moreover, O-ViT uses no hard orthogonality constraint.
3. We conduct comparative experiments between O-ViT and ViT on well-known datasets, which demonstrate the superiority of O-ViT over other existing ViTs.

This rest of this paper is organized as follows. After Section 2 introduces the background, Section 3 details the framework and parameterization strategy of O-ViT. Section 4 presents experimental results to show the superiority of O-ViT. Finally, Section 5 concludes the paper.

2 BACKGROUND

The proposed O-ViT combines ViT and the optimization on manifold for deep learning.

2.1 VISION TRANSFORMER (VIT)

Based on the assumption of translation invariance Touvron et al. (2021b); Kayhan & van Gemert (2020), CNN shares and translates one convolution kernel filter to extract local features at different positions in one channel. To get rid of CNN, ViT takes advantage of the self-attention mechanism, the essence of which is represented as

$$\begin{aligned} &Q, K, V = XW_Q, XW_K, XW_V, \\ &Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \end{aligned} \tag{1}$$

where X is input feature. W_Q, W_K, and W_V are trainable matrices applied to X to generate query matrix Q, key matrix K, and value matrix V. d_k is the dimension of K. The self-attention block measures the correlation between different projection spaces (Q and K), and the normalized correlation is applied to V as an attention map. ViT first embeds the input image into fixed-size patches and then embeds their positional information named Patch Embedding and Positional Embedding

Fayyaz et al. (2021). Then, the scaled dot-product self-attention mechanism, which is served as an encoder, is applied to the above embedding. Equation (1) is also called single-head self-attention, which can be improved by multi-head ones Yan et al. (2019), i.e.,

$$\begin{aligned} head^{(h)} &= Attention(Q^{(h)}, K^{(h)}, V^{(h)}) \\ MultiHead(X) &= [head^{(1)}; \cdots ; head^{(n)}]W_O, \end{aligned} \tag{2}$$

where n is the number of heads, h is the head index, and $[head^{(1)}; \cdots ; head^{(n)}]$ means concatenating n heads in the last dimension. Let $d = n \times d_k$, W_O is a learnable parameter of size $\mathbb{R}^{d \times d}$.

ViT can be interpreted by a biologically plausible memory model named Sparse Distributed Memory (SDM) Bricken & Pehlevan (2021). The intersection of hyperspheres adopted by read operations in SDM can approximate the softmax function in ViT. Although it has no relation with orthogonality, such geometry interpretation inspires us to rethink ViT from a geometric view. There are various variants of ViT, such as deeper ViTs Touvron et al. (2021b); Zhou et al. (2021), compact transformers Wu et al. (2021a), cross transformers processing image at different scales Chen et al. (2021), and twin transformers mixing local and global attention Chu et al. (2021). Orthogonal parameterization does not conflict with the above variants and can be applied to them as a convenient plug-and-play.

2.2 Optimization on Manifold for Deep Learning

Let H^n indicate the half space defined by $x_1 \geq 0$ in n-dimensional Euclidean space R^n. Hausdorff space M is called n-dimensional topological manifold when each point p has an open neighborhood $U(p)$ homeomorphic with R^n or H^n. There are two steps in optimization on manifolds: orthogonal projection and retraction operation Bronstein et al. (2021). As shown in Figure 1, on a manifold $\mathcal{M}$, $f(\theta)$ descents steepest in the direction of $\mathbf{H}$, which is opposite to the direction of Riemannian gradient $grad\ f(\theta)$ Hawe (2013). $grad\ f(\theta)$ can be obtained by orthogonal projection Π, which projects the gradient at a point θ from ambient Euclidean space to tangent space $T_\theta\mathcal{M}$:

$$grad\ f(\theta) = \Pi_{T_\theta\mathcal{M}}(\nabla f(\theta)), \tag{3}$$

where $\nabla f(\theta)$ represents the Euclidean gradient. The smooth red curve in Figure 1 denotes a geodesic $\Gamma_\theta(\gamma\mathbf{H})$ in the direction of $\mathbf{H}$ with a step size γ. The geometric optimization requires to update point θ to point θ' along the curve $\Gamma_\theta(\gamma\mathbf{H})$, in an opposite direction of $grad\ f(\theta)$. Due to high complexity, the geodesic is approximated by the retraction $\mathbb{R}_\theta(\gamma\mathbf{H}) : T_\theta\mathcal{M} \to \mathcal{M}$ in practice Kumar et al. (2018), mapping updated parameters from the tangent space back to the manifold.

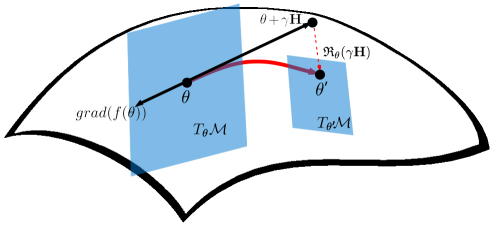


Figure 1: Update θ to θ' on a smooth manifold

3 Our Proposed Method

We pay attention to the invariant metric inherited by orthogonal matrices and creatively restrict linear transformation matrices of self-attention in ViT to reside on the orthogonal manifold. We also explore a computationally economic way to parameterize them. We first briefly describe O-ViT's architecture in Section 3.1. Then we introduce O-ViT's orthogonality technique in Section 3.2 and explain theoretical advantages that support its efficiency in optimization in Section 3.3.

3.1 O-ViT Architecture

O-ViT architecture differs from other ViTs in the design of the self-attention block. Given the input X, O-ViT defines an orthogonal self-attention block as

$$Q, K, V = X\,h(A^Q), X\,h(A^K), X\,h(A^V), \tag{4}$$

where A^Q, A^K and A^V are all skew-symmetric matrices Lee (2005), and they can be extended to skew-Hermitian matrices in case of unitary constraints. Algorithm 1 and Algorithm 2 present the orthogonalization performed over self-attention block. Algorithm 1 demonstrates three possible strategies of orthogonal parameterization (e.g., Skew Symmetric Transited Orthogonalization ("SSTO"), "Exp" and "Cayley"). Lines 1-2 shows the "Exp" orthogonalization that utilizes the Lie exponential mapping on matrix Lie groups, i.e., $exp(W) := E + W + \frac{W^2}{2} + \cdots$. Our "SSTO" orthogonalization adopts a two-step strategy. Firstly, line 4 transforms an arbitrary weight matrix to a skew-symmetric one. Then line 5 employs $h(W) = 2(E + W)^{-1} - E$ to map it to the orthogonal group. There is a special relationship between skew-symmetric and orthogonal matrices, therefore, we use skew-symmetric matrices as a transition to realize orthogonal constraints. It will be detailed in the following subsection. Lines 6-7 present the "Cayley" orthogonalization, $[(E + \frac{W}{2})(E - \frac{W}{2})]^{-1}$, which is a first order approximation of the Riemannian exponential mapping. As seen in the Algorithm 2, the query, key, and value weight matrices are imposed orthogonal parameterization (refer to lines 1-3) before projecting input feature X to corresponding query, key and value spaces (refer to line 4) and calculating the attention map (refer to line 5).

Algorithm 1 Parameterization

Input: W, orthType
Output: W

1: **if** orthType == "Exp" **then**
2: $W = matrix\_exponential(W) = \Sigma_{k=0}^{\infty} \frac{W^k}{W!}$
3: **else if** orthType == "SSTO" **then**
4: $W = W - W^T$
5: $W = 2(E + W)^{-1} - E$
6: **else if** orthType == "Cayley" **then**
7: $W = [(E + \frac{W}{2})(E - \frac{W}{2})]^{-1}$
8: **end if**
9: **return** W

Algorithm 2 Orth_Self_Attention

Input: X, orthType
Parameter: W_Q, W_K, W_V
Output: attn

1: W_Q = Parameterization(W_Q, orthType)
2: W_K = Parameterization(W_K, orthType)
3: W_V = Parameterization(W_V, orthType)
4: $Q, K, V = XW_Q, XW_K, XW_V$
5: $attn = softmax(\frac{QK^T}{\sqrt{d_k}})V$
6: **return** attn

Furthermore, Algorithm 3 presents our architectural innovation for self-attention block, i.e., our MultiOrth self-attention block introduces an aggregation layer to synthesize the attention driven by multiple orthogonal parameterization strategies. "X" and "attn" represent the input and output of MultiOrth self-attention block, respectively. "orthoType" indicates the orthogonal implementation type of the aggregation layer, whose value is in the set $\{$"$Cayley$", "$SSTO$", "Exp"$\}$. Lines 1-3 reveal that all of three orthogonalization strategies are adopted. The basis of orthogonal parameter space formed by different orthogonalization strategies are different, learning differing components of the attention map. To comprehensively accept the different attention value, this paper applies a linear layer as an aggregator on top of the concatenation of them (refer to lines 4-6 and line 10). Along with training, elements of the linear layer will converge to weight or vote for three orthogonal self-attention blocks appropriately. Furthermore, we can even restrict parameters of aggregation layer to reside on the orthogonal manifold (refer to lines 7-9), when countering complex variants of ViT that may exacerbate feature redundancy. Note that the parameterization in Algorithm 2 is not an initialization. It will be executed in every forward epoch of the orthogonal self-attention block.

Algorithm 3 MultiOrth_Self_Attention

Input: X, orthType
Output: attn

attnExp = Orth_Self_Attention(X, "Exp")
attnSSTO = Orth_Self_Attention(X, "SSTO")
attnCayley = Orth_Self_Attention(X, "Cayley")
attnConcat = [attnExp: attnSSTO: attnCayley]
n = attnConcat.last_dimension
Aggregator = Linear_Layer(input_dim = n, output_dim = $\frac{1}{3}$n)
if orthType != null and orthType in ["Exp", "SSTO", "Cayley"] **then**
    Aggregator.weight
        = Parameterization(Aggregator.weight, orthType)
end if
attn = Aggregator(attnConcat)
return attn

3.2 Orthogonal Parameterization

O-ViT employs skew-symmetric matrices Casado & Martínez-Rubio (2019) as a transition to realize orthogonal constraints. The Lie algebras of special orthogonal group Casado & Martínez-Rubio (2019) and unitary group are Casado & Martínez-Rubio (2019) skew-symmetric and skew-Hermitian matrices. They are isomorphic to a vector space Casado & Martínez-Rubio (2019). Any real square matrix $A \in \mathbb{R}^{n\times n}$ can be mapped into a skew-symmetric matrix by $A - A^T$, given that $(A - A^T) + (A - A^T)^T = 0$. In the same token, any complex square matrix $A \in \mathbb{C}^{n\times n}$ can be transformed into a skew-Hermitian matrix.

In the Lie group theory, the exponential mapping $exp : g \to G$ Casado & Martínez-Rubio (2019) builds correspondence between skew-symmetric matrices $\mathfrak{so}(n)$ and its Lie Group $O(n)$. Though the mapping $exp(X) := \Sigma_{k=0}^{\infty} \frac{X^k}{k!}$ is not surjective in general, compact Lie groups are one of special families in which the exponential mapping is surjective. However, it is computationally expensive, and the huge number produced by the exponent may induce gradient vanishing problems in the softmax function. As an alternative solution, Cayley method is the first order approximation. Moreover, this paper focuses on the advantages of orthogonality over ViT models. In order to fully demonstrate the merits of the orthogonality over ViT, various kinds of orthogonalization implementations are necessary to be investigated. As a supplement, we use the map $h : g \to G$, $h := 2(E+X)^{-1}-E$ Hsu (1953) to project any skew-symmetric matrix $X \in \mathbb{R}^{n\times n}$ to the orthogonal group and name the above projection as "SSTO". Moreover, the equation $h(X) = 2(E + X)^{-1} - E$ is a surjective mapping between the orthogonal group and its Lie algebra Hsu (1953) . For any $Y \in O(n)$, there exists an skew-symmetric matrix X that satisfies $h(X) = Y$.

Proof. $h(X)h^T(X) = [2(E+X)^{-1} - E][2(E+X)^{-1} - E]^T = [2(E+X)^{-1} - (E+X)^{-1}(E+X)][2(E+X)^{-1} - (E+X)^{-1}(E+X)]^T = (E+X)^{-1}(E+X) = E.$

$\forall Y \in O(n)$, we have $X = 2(E+Y)^{-1} - E$ Hsu (1953) satisfies: $h(X) = h(2(E+Y)^{-1} - E) = 2[E + [2(E+Y)^{-1} - E]]^{-1} - E = Y.$ □

3.3 From Riemannian to Euclidean Optimization

Manifold optimization belongs to the domain of constrained optimization Kotary et al. (2021), therefore, the constraint indicated by specific Riemannian manifold should be satisfied in the minimization of the optimization objective. To achieve this, the optimal solution must be searched on Riemannian manifolds rather than Euclidean space, which requires orthogonal projection and retraction operation. The above steps do not exist in Euclidean optimization, which makes general optimizers useless in geometric optimization. To address this problem, Kumar et al. (2018) introduced constraint Stochastic Gradient Descent-Momentum (SGD-M) and constraint Root Mean Square Prop (RMSProp) as a counterpart of regular optimizers in Euclidean space. However, considering that it is an expensive computation overhead to operate orthogonal projection and retraction, O-ViT chooses to avoid them rather than adapt to them. Our O-ViT's parameterization has the following properties that helps it gain the above goal and become a sensible option for geometry optimization.

The optimization of O-ViT can be transformed into an optimization problem in Euclidean space. Let θ_B represent the trainable parameter subjected to the orthogonal group, the constrained optimization problem defined as follows

$$\min_{\theta_B \in G} f(x; \theta_B) \tag{5}$$

is equivalent to following optimization problem, i.e.,

$$\min_{\theta_A \in g} f(x; \theta_A), \tag{6}$$

where θ_A is a skew-symmetric matrix. Evidently, an optimal solution $\hat{\theta_B}$ for Equation (5) and an optimal solution $\hat{\theta_A}$ for Equation (6) have an equivalent relationship that $\hat{\theta_B} = h(\hat{\theta_A})$, since the map $h : g \to G$ introduced in Section 3.2 is surjective. Therefore, if the second problem has a solution, then we will definitely find a solution to the first problem. As a result, our O-ViT can be optimized with Euclidean optimizers. Since the skew-symmetric matrix space is isomorphic to a vector space, Equation (6) is actually a non-constrained problem. As described in Figure 1, $\Delta\theta_B = -\gamma \; grad \; f(x; \theta_B)$ is on the tangent space $T_{\theta_B} M$, rather than along the geodesic curve Hawe (2013). Therefore, a retraction $\mathbb{R}_\theta(\Delta\theta_B)$ from tangent space to manifold is needed, and θ_B should be updated by $\theta_B \mathbb{R}_\theta(\Delta\theta_B)$ rather than $\theta_B + \Delta\theta_B$, i.e.,

$$\theta'_B \leftarrow \theta_B \mathbb{R}_\theta(-\gamma \; grad \; f(x; \theta_B)). \tag{7}$$

Moreover, the mapping $\hat{\theta_B} = h(\hat{\theta_A})$ induces the following iteration of trainable parameter θ_A

$$h(\theta'_A) \leftarrow h(\theta_A - \gamma \nabla (f \circ h)(x; \theta_A)), \tag{8}$$

where the gradient $\nabla(f \circ h)$ is defined in Euclidean space, making trainable parameters of Equation (6) updated in Euclidean space. As a consequence, traditional gradient descent optimizers such as ADAM can be directly used to optimize the orthogonality-constrained O-ViT. Furthermore, our O-ViT does not create saddle points. Saddle points are unstable fixed points of the gradient descent optimization algorithm and are difficult to reach on the orthogonal manifold Absil et al. (2008). $h(X) = 2(E + X)^{-1} - E$ constructs a one-to-one correspondence between the skew-symmetric matrices and the orthogonal group. Provided that the optimization problem stays in its tangent space $\mathfrak{o}(n)$, the parameter update is unique. It implies that our parameterization avoids saddle points.

4 EXPERIMENTS

To evaluate the efficiency of our proposed O-ViT, we conducted comparative experiments between O-ViT and ViT on different datasets. We assessed the performance of O-ViT in three aspects: i) under same conditions, the recognition accuracy of O-ViT is higher than ViT and the convergence of the O-ViT is faster, ii) O-ViT withstands the disturbance of noise better than ViT, and iii) O-ViT can reduce the numbers of parameters while ensuring a credible accuracy.

We used three benchmarks: BaseViT, DeepViT, and CaiT, whose license is MIT license. BaseViT means the most original and fundamental ViT. DeepViT Zhou et al. (2021), and CaiT Touvron et al. (2021b) are variants of BaseViT, therefore, we used them as a supplement to the original ViT. We are the first to use exponential mapping (Exp) Lezcano-Casado (2019) on ViT. ViTs orthogonalized by exponential mapping, skew symmetric transited orthogonalization, and cayley strategies can be collectively referred to as Exp-ViTs (e.g., Exp-BaseViT, Exp-DeepViT, and Exp-CaiT), SSTO-ViTs, and Cayley-ViTs. All of the above ViTs belong to O-ViT, and can be referred to as SingleOrth-ViT.

We implemented our O-ViT on top of the deep learning framework PyTorch. Unless otherwise stated, the reported results were measured in Top-1 Accuracy, and we did not take the Top-5 Accuracy into consideration. We set the same cropped size 32×32 for input data (except for the ImageNet dataset cropped into a size of 224×224) and the same hyper-parameters for the neural network for the fair comparison in one control group. We employed a standard data augmentation strategy: random rotation, crop, and horizontal flip. We used AdaGrad as the optimizer. The learning rate was set as 5.0×10^{-3} initially, whose value would be dynamically adjusted during training. We set weight decay and momentum as 7.0×10^{-4} and 0.9, respectively. We performed experiments on PCs with a single Nvidia GTX 3090 GPU.

TABLE 1: TOP1-ACCURACY COMPARISON OF RECOGNITION PROBLEMS

Method	SVHN	YaleB	CIFAR10	CIFAR100	Caltech101	ImageNet50
BaseViT	94.65%	95.94%	82.79%	56.27%	47.46%	47.12%
Exp-BaseViT (O-VIT)	**95.72%**	97.32%	85.40%	58.75%	47.34%	46.32%
SSTO-BaseViT (O-VIT)	94.47%	98.13%	83.65%	57.16%	45.59%	47.25%
Cayley-BaseViT (O-VIT)	95.36%	99.25%	84.43%	58.21%	48.42%	46.56%
MO-BaseViT (O-VIT)	95.60%	**99.52%**	**86.07%**	**61.65%**	**53.90%**	**48.40%**
DeepViT	21.59%	72.32%	60.51%	34.89%	47.34%	32.88%
Exp-DeepViT (O-VIT)	83.33%	94.82%	64.08%	36.43%	46.78%	32.72%
SSTO-DeepViT (O-VIT)	85.43%	99.57%	63.81%	36.63%	52.82%	**35.12%**
Cayley-DeepViT (O-VIT)	82.66%	97.12%	63.65%	36.55%	51.86%	32.68%
MO-DeepViT (O-VIT)	**87.86 %**	**99.73%**	**65.88%**	**37.32%**	**54.52%**	33.52%
CaiT	91.97%	12.01%	75.39%	49.50%	25.99%	47.20%
Exp-CaiT (O-VIT)	**94.99%**	88.31%	82.72%	57.01%	50.51%	50.36%
SSTO-CaiT (O-VIT)	92.32%	99.57%	79.03%	52.00%	54.80%	**56.00%**
Cayley-CaiT (O-VIT)	94.03%	99.68%	80.06%	55.59%	54.07%	50.24%
MO-CaiT (O-VIT)	93.67%	**99.79%**	**83.79%**	**63.91%**	**55.59%**	53.28%

[1] Baselines of "BaseViT" and "CaiT" are downloaded from https://github.com/kentaroy47/vision-transformers-cifar10. Baselines of "DeepViT" are downloaded from https://github.com/lucidrains/vit-pytorch.

[2] "MO" combines all of three orthogonalization strategies. In "MO-DeepViT" and "MO-CaiT", parameters of aggregation layer are orthogonalized by "SSTO".

4.1 ABLATION STUDY

We chose various image recognition tasks (e.g., character recognition task, face recognition task, and object recognition task) to evaluate the performance of O-ViT in comparison with ViT. Table 2 presents details such as the number of categories, and the size of training set and test set. ImageNet is such a huge article classification dataset that there are altogether 1000 categories in it. We selected first 50 categories as a subset for experiments named ImageNet50. Note that the data we used does not contain personally identifiable information or offensive content.

TABLE 2: VIT BENCHMARK CONFIGURATIONS

Dataset	Training Set #	Testing Set #	Categories #
SVHN	73257	26032	10
YaleB	2314	1874	38
CIFAR10	50000	10000	10
CIFAR100	50000	10000	100
Caltech101	7280	1864	102
Imagenet50	25000	5000	50

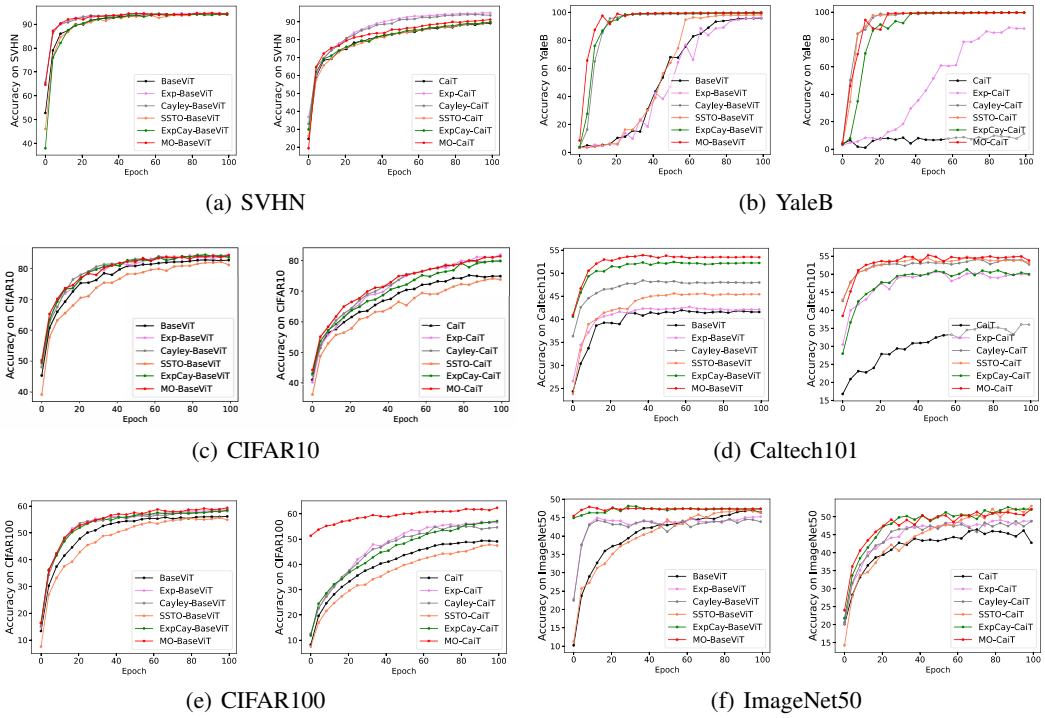
We want to figure out two issues in the ablation stage: i) what is the efficiency of O-ViT compared with ViT, and ii) whether an aggregation of different orthogonal strategies works better than single orthogonalization. Moreover, we selected DeepViT and CaiT to show the efficiency of O-ViT on deepening the network architecture, since DeepViT and CaiT involve more than one attention block. All results were obtained by training for 100 epochs from scratch. Table 3 presents details of ViT benchmarks. We set the patch size to be 16×16 for ImageNet50 and 4×4 for other datasets.

TABLE 3: VIT BENCHMARK CONFIGURATIONS

Parameters	BaseViT	DeepViT	CaiT
Self-Attention Block #	1	6	9
Hidden size	512	512	512
MLP size	2048	2048	2048
Heads	12	8	8

Figure 2 plots the classification accuracy vs. epoch for different methods on datasets from SVHN to ImageNet50 (please refer to Section A.1 for the full version). Almost all O-ViTs outperform ViTs in terms of classification accuracy and convergence speed (65 out of 72), which illustrates that introducing orthogonal constraints in self-attention blocks can improve the visual performance of ViT. For example, by imposing multi-orthogonal parameterization, the accuracy of BaseViT is improved by 6.4% on the Caltech101 dataset.

Moreover, we pay attention to the influence of orthogonal constraints on the depth of ViT. Since both DeepViT and CaiT are explorations of deepening ViT, the success of imposing orthogonal



(a) SVHN (b) YaleB

(c) CIFAR10 (d) Caltech101

(e) CIFAR100 (f) ImageNet50

Figure 2: Performance comparison for different datasets

constraints on them shows the potential of orthogonal parameterization on increasing the depth of the network. The self-attention mechanism adopts the softmax function to normalize the similarity between the query and key, while exponents in softmax induce zero gradients resulted from very large numbers. When the zero gradient is transmitted to front layers, the shrinking effects will grow exponentially, yielding the gradient vanishing problem. Parameters are updated in accordance with the direction of gradient descent, hence, the vanishing gradient will inevitably restrict ViT to go deeper. Orthogonal parameterization can alleviate the above gradient vanishing problem due to its norm-keeing property, helping ViT go deeper.

MultiOrth-ViT outperforms SingleOrth-ViT (e.g., Exp-ViT, SSTO-ViT and Cayley-ViT) in most cases (45 out of 54). As seen from Table 1 (please refer to Section A.2 for the full version), the effectiveness of aggregating multiple orthogonalization strategies on learning a more essential feature map is reflected incisively and vividly in DeepViT (17 out of 18). Although all of the three orthogonal strategies can learn more essential and non-redundant features, they restrict orthogonal parameters in different coordinate systems and learn different sides. Therefore, combining different orthogonal strategies will learn more comprehensive features. On account of 6 or 9 self-attention blocks, DeepViT and CaiT have a lot more parameters than BaseViT. To mitigate over-fitting, we orthogonalize parameters in the aggregation layer. Experimental results confirm that the above practice can enhance the recognition accuracy.

4.2 Robustness

To evaluate the robustness of O-ViTs compared to ViTs, we added four kinds of noise to different datasets' testing samples. O-ViTs we chose in this stage belong to MO-ViTs. All noises obey the Gaussian distribution with the expected value of 0. Let std represent the standard deviation of the Gaussian distribution, four Gaussian noises were: i) $std = 0.05$, ii) $std = 0.08$, iii) $std = 0.1$ and iv) $std = 1$. We only show two kinds of noise interference (refer to Table 4) since space is limited. Please see the appendix Section A.3 for the full version. We employed the recognition accuracy of noise-corrupted images to measure the robustness of methods. Consequently, the higher value of accuracy represents the stronger robustness.

Table 4: Comparison between O-ViTs and ViTs with Noises

Method	YaleB		CIFAR10		Caltech101		ImageNet50	
	std = 0.1	std = 0.05	std = 0.1	std = 0.05	std = 0.1	std = 0.05	std = 0.1	std = 0.05
BaseViT	65.15%	94.88%	82.75%	82.79%	**36.95%**	**45.42%**	**45.21%**	**46.92%**
O-BaseViT	**74.07%**	**96.37%**	**85.96%**	**86.04%**	33.73%	36.84%	45.02%	45.64%
DeepViT	64.30%	70.92%	55.16%	59.67%	41.92%	43.56%	30.72%	31.68%
O-DeepViT	**96.16%**	**99.04%**	**56.11%**	**60.65%**	**51.41%**	**52.09%**	**34.08%**	**34.72%**
CaiT	3.63%	3.74%	75.09%	75.42%	25.25%	25.42%	43.0%	43.28%
O-CaiT	**98.72%**	**99.31%**	**82.66%**	**82.50%**	**53.22%**	**54.46%**	**50.61%**	**51.35%**

Table 4 shows the comparison between robustness performance between ViTs and O-ViTs on YaleB, CIFAR10, Caltech101, and ImageNet50 datasets with noises at different intensities. We can see that methods with O- prefixes outperform their counterparts in most cases (20 out of 24) considering noise. The above results confirm that orthogonal projections can resist the corruption of input images to a certain extent, which makes O-ViT have stronger robustness than ViT. For example, for the CIFAR10 dataset, We can see a sharp increase in the robustness performance of CaiT after imposing orthogonal constraints. Moreover, the O-DeepViT shows obvious advantages over its non-orthogonal counterpart on YaleB and Caltech101 dataset. Table 4 also presents that, for other datasets with noise corruption, OViTs perform better ViTs at least 1% and up to two times.

To sum up, methods applied orthogonal constraints (O-ViTs) yield a higher recognition accuracy in majority cases with noise turbulence, which confirms the stronger robustness of orthogonal parameterization compared with general parameterization under noises.

Table 5: Comparison between O-ViT and ViT in Terms of Accuracy and the Number of Parameters

Dataset	ViT		O-ViT	
	accuracy [%]	parameters [M]	accuracy [%]	parameters [M]
YaleB	95.94	19.01	98.15	10.74
SVHN	94.65	19.07	94.29	12.77
CIFAR10	82.79	19.07	80.35	9.62
CIFAR100	56.27	19.12	55.41	9.67
Caltech101	47.46	19.23	47.97	10.92
ImageNet50	47.12	21.01	45.76	13.70

4.3 The Number of Parameters

Table 5 shows recognition accuracy and the number of trainable parameters of O-ViT and ViT on different datasets. O-ViTs we chose in this stage are parameterized by skew symmetric transited orthogonalization. On the YaleB and Caltech101 dataset, O-ViT recognizes more accurately than ViT while the number of parameters of O-ViT is nearly half of ViT with the same depth. As to other datasets, O-ViT is less accurate than ViT by a narrow margin while the number of parameters of O-ViT is significantly smaller than that of ViT with the same depth. Orthogonal parameters can reduce redundancy theoretically, while the above experiment results confirm that O-ViT can reduce the number of parameters while guaranteeing an acceptable accuracy.

5 Conclusion

Self-attention in ViT performs well on image recognition tasks. However, its efficiency still has space for investigation and development. In this study, low intra-similarity in the orthogonal matrix and metric invariance property of orthogonal transformations were concerned. We imposed orthogonal constraints on ViT and proposed a novel approach, O-ViT, to push the boundaries of the existing ViT in a geometric way. Moreover, we utilized an implementation trick based on classic Lie group theory to simplify the constrained optimization over compact Lie groups, e.g., the orthogonal group. It is of independent interest and could have more applications in variants of ViT or in combination with other machine learning methods. Furthermore, we have conducted comparative experiments on different vision recognition tasks to provide practical evidence of O-ViT's performance. Experiments also proved the soundness of O-ViT in deepening the self-attention in ViT.

REFERENCES

P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. 2008.

Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary Evolution Recurrent Neural Networks. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 1120–1128, 2016.

Trenton Bricken and Cengiz Pehlevan. Attention Approximates Sparse Distributed Memory. *ArXiv*, abs/2111.05498, 2021.

Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Velivckovi'c. Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges. *ArXiv*, abs/2104.13478, 2021.

Mario Lezcano Casado and David Martínez-Rubio. Cheap Orthogonal Constraints in Neural Networks: A Simple Parametrization of the Orthogonal and Unitary Group. *ArXiv*, abs/1901.08428, 2019.

Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification, 2021.

Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting spatial attention design in vision transformers, 2021.

Mohsen Fayyaz, Soroush Abbasi Kouhpayegani, Farnoush Rezaei Jafari, Eric Sommerlade, Hamid Reza Vaezi Joze, Hamed Pirsiavash, and Juergen Gall. Ats: Adaptive token sampling for efficient vision transformers. *arXiv preprint arXiv:2111.15667*, 2021.

Simon Alois Hawe. *Learning Sparse Data Models via Geometric Optimization with Applications to Image Processing*. PhD thesis, Universität München, 2013.

P. L. Hsu. On symmetric, orthogonal, and skew-symmetric matrices. *In Proc. of Edinburgh Mathematical Society*, 10(1):37–44, 1953.

Z. Huang and L. Gool. A riemannian network for spd matrix learning. In *Proc. of AAAI Conf. on Artificial Intelligence (AAAI)*, pp. 2036–2042, 2017.

Zhiwu Huang, Jiqing Wu, and Luc Van Gool. Building deep networks on grassmann manifolds. In *Proc. of AAAI Conf. on Artificial Intelligence (AAAI)*, volume 32, 2018.

Osman Semih Kayhan and Jan C. van Gemert. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 14262–14273, 2020.

Alexander Kolesnikov, Alexey Dosovitskiy, Dirk Weissenborn, Georg Heigold, Jakob Uszkoreit, Lucas Beyer, Matthias Minderer, Mostafa Dehghani, Neil Houlsby, Sylvain Gelly, Thomas Unterthiner, and Xiaohua Zhai. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. of Intl. Conf. on Learning Representations (ICLR)*, 2021.

James Kotary, Ferdinando Fioretto, Pascal Van Hentenryck, and Bryan Wilder. End-to-end constrained optimization learning: A survey. In *In Proc. of Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, pp. 4475–4482, 2021.

S. Kumar, Z. Mhammedi, and M. Harandi. Geometry aware constrained optimization techniques for deep learning. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 4460–4469, 2018.

Tarek A. Lahlou and Alan V. Oppenheim. Trading accuracy for numerical stability: Orthogonalization, biorthogonalization and regularization. In *Proc. of ICASSP*, pp. 4747–4751, 2016.

Pei Yean Lee. *Geometric optimization for computer vision*. PhD thesis, The Pennsylvania State University, 2005.

Mario Lezcano-Casado. Trivializations for gradient-based optimization on manifolds. In *Proc. of Advances in Neural Information Processing Systems (NeurIPS)*, pp. 9154–9164, 2019.

P Rodríguez, J Gonzàlez, G. Cucurull, J. M. Gonfaus, and X. Roca. Regularizing cnns with locally constrained decorrelations. 2017.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *Arxiv*, abs/1312.6120, 2014.

Steven T Smith. Optimization techniques on Riemannian manifolds. *Fields Institute Communications*, 3(3):113–135, 1994.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient Image Transformers & distillation through attention. In *Proc. of International Conference on Machine Learning (ICML)*, pp. 10347–10357, 2021a.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, and Gabriel Synnaeve. Going deeper with Image Transformers. In *Proc. of International Conference on Computer Vision (ICCV)*, 2021b.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. of Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, 2017.

Jiayun Wang, Yubei Chen, Rudrasis Chakraborty, and Stella X Yu. Orthogonal convolutional neural networks. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 11505–11515, 2020.

Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers, 2021a.

Xia Wu, Xueyuan Xu, Jianhong Liu, Hailing Wang, Bin Hu, and Feiping Nie. Supervised feature selection with orthogonal regression and feature weighting. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):1831–1838, 2021b.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. Tener: adapting transformer encoder for named entity recognition. *arXiv preprint arXiv:1911.04474*, 2019.

Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis E.H. Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-Token ViT: Training Vision Transformers From Scratch on Imagenet. In *Proc. of International Conference on Computer Vision (ICCV)*, 2021.

Aston Zhang, Alvin Chan, Yi Tay, Jie Fu, Shuohang Wang, Shuai Zhang, Huajie Shao, Shuochao Yao, and Roy Ka-Wei Lee. On orthogonality constraints for transformers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 375–382, 2021.

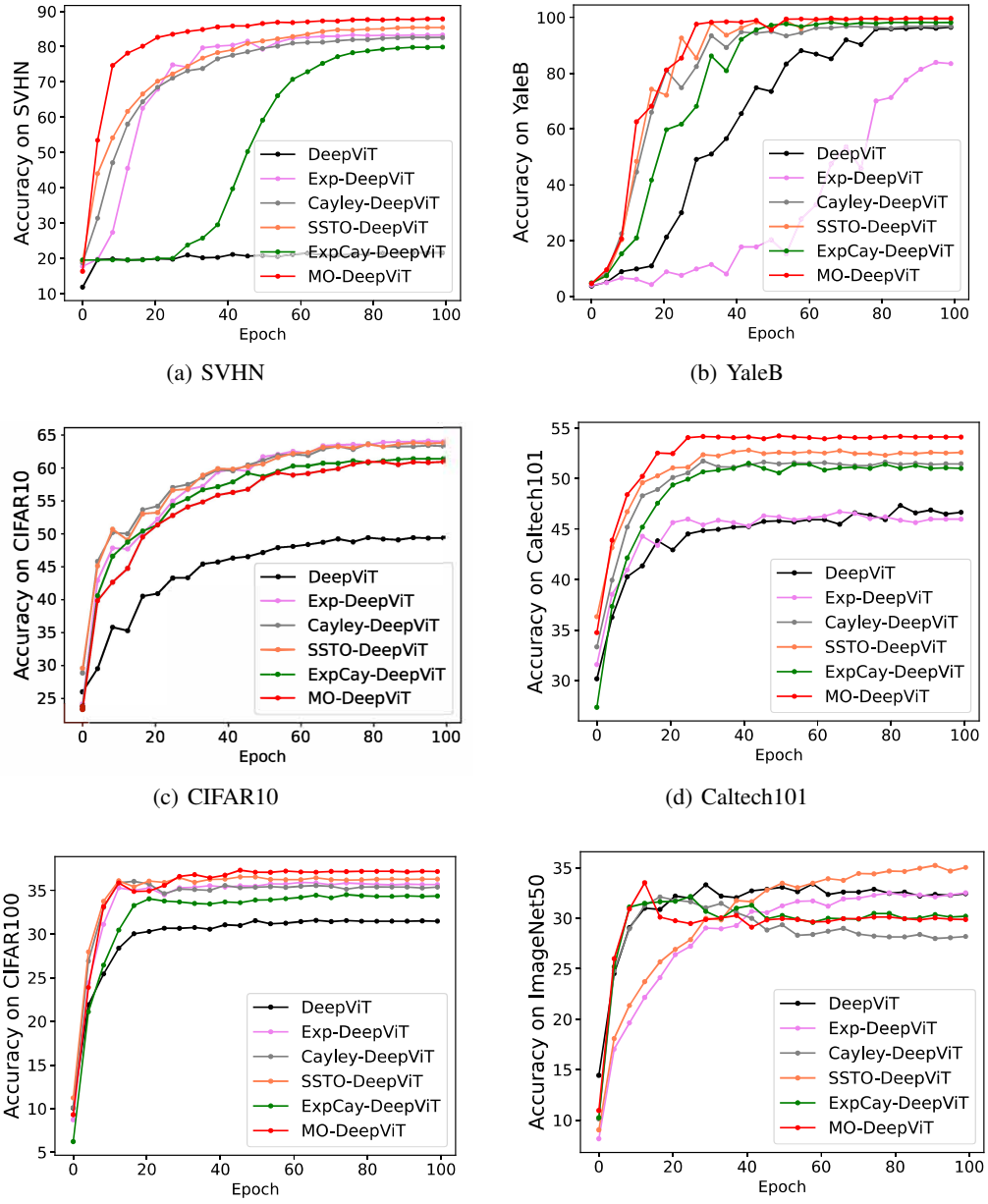
Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021.

J. Zhou, M. N. Do, and J Kovačević. Ieee transactions on image processing 1 special paraunitary matrices, cayley transform, and multidimensional orthogonal filter banks. *IEEE Transactions on Image Processing*, 14(6):760, 2008.

A APPENDIX

A.1 PERFORMANCE COMPARISON FOR DIFFERENT DATASETS

Figure 1 shows the full version of the performance comparison for different datasets.



(a) SVHN (b) YaleB

(c) CIFAR10 (d) Caltech101

(e) CIFAR100 (f) ImageNet50

Figure 1: Performance comparison for different datasets

A.2 TOP1-ACCURACY COMPARISON RESULTS OF RECOGNITION PROBLEMS

Table 1 shows the full version of top1-accuracy comparison results of different recognition problems between O-ViT and ViT.

TABLE 1: TOP1-ACCURACY COMPARISON RESULTS OF RECOGNITION PROBLEMS

Method	SVHN	YaleB	CIFAR10	CIFAR100	Caltech101	ImageNet50
BaseViT	94.65%	95.94%	82.79%	56.27%	47.46%	47.12%
Exp-BaseViT (O-ViT)	**95.72%**	97.32%	85.40%	58.75%	47.34%	46.32%
SSTO-BaseViT (O-ViT)	94.47%	98.13%	83.65%	57.16%	45.59%	47.25%
Cayley-BaseViT (O-ViT)	95.36%	99.25%	84.43%	58.21%	48.42%	46.56%
ExpCay-BaseViT (O-ViT)	94.65%	**99.63%**	85.77%	60.40%	52.43%	**48.60%**
MO-BaseViT (O-ViT)	95.60%	99.52%	**86.07%**	**61.65%**	**53.90%**	48.40%
DeepViT	21.59%	72.32%	60.51%	34.89%	47.34%	32.88%
Exp-DeepViT (O-ViT)	83.33%	94.82%	64.08%	36.43%	46.78%	32.72%
SSTO-DeepViT (O-ViT)	85.43%	99.57%	63.81%	36.63%	52.82%	**35.12%**
Cayley-DeepViT (O-ViT)	82.66%	97.12%	63.65%	36.55%	51.86%	32.68%
ExpCay-DeepViT (O-ViT)	79.94%	98.51%	61.40%	34.52%	51.69%	32.28%
MO-DeepViT (O-Lin) (O-ViT)	**87.86 %**	**99.73%**	**65.88%**	**37.32%**	**54.52%**	33.52%
CaiT	91.97%	12.01%	75.39%	49.50%	25.99%	47.20%
Exp-CaiT (O-ViT)	**94.99%**	88.31%	82.72%	57.01%	50.51%	50.36%
SSTO-CaiT (O-ViT)	92.32%	99.57%	79.03%	52.00%	54.80%	**56.00%**
Cayley-CaiT (O-ViT)	94.03%	99.68%	80.06%	55.59%	54.07%	50.24%
ExpCay-CaiT (O-ViT)	91.71%	99.76%	82.68 %	59.79%	51.69%	54.18%
MO-CaiT (O-Lin) (O-ViT)	93.67%	**99.79%**	**83.79%**	**63.91%**	**55.59%**	53.28%

[1] Both "ExpCay" and "MO" belong to "MultiOrth-ViT". "ExpCay" is the combination of "Exp" and "Cayley". "MO" is the combination of all of three orthogonalization strategies. "O-Lin" indicates that parameters of aggregation layer are on the orthogonal manifold, orthogonalized by "SSTO".

A.3 COMPARISON BETWEEN O-VITS AND VITS CONSIDERING NOISES

Table 2, Table 3 and Table 4 show the robustness comparison between O-ViTs and ViTs considering four different kinds of noises, respectively.

TABLE 2: COMPARISON BETWEEN O-VITS AND VITS WITH NOISES

Method	SVHN				CIFAR10			
	std = 1	std = 0.1	std = 0.08	std = 0.05	std = 1	std = 0.1	std = 0.08	std = 0.05
BaseViT	94.93%	94.98%	94.98%	94.98%	82.79%	82.75%	82.85%	82.79%
O-BaseViT	**95.60%**	**95.96 %**	**96.02 %**	**96.04%**	**85.78%**	**85.96%**	**86.30 %**	**86.04%**
DeepViT	**19.50%**	19.61%	19.58%	19.56%	11.09%	55.16%	57.29%	59.67%
O-DeepViT	10.38 %	**74.40%**	**78.16%**	**82.31%**	**11.11%**	**56.11%**	**58.49%**	**60.65%**
CaiT	86.31%	86.37%	86.28%	86.33%	74.67%	75.09%	75.04%	75.42%
O-CaiT	**87.19%**	**87.06%**	**87.19%**	**87.33%**	**82.12%**	**82.66%**	**82.13 %**	**82.50%**

Table 3: Comparison between O-ViTs and ViTs with Noises

Method	CIFAR100				Caltech101			
	std = 1	std = 0.1	std = 0.08	std = 0.05	std = 1	std = 0.1	std = 0.08	std = 0.05
BaseViT	55.78%	56.21%	56.21%	56.29%	1.13%	**36.95%**	**41.07%**	**45.42%**
O-BaseViT	**57.97%**	**58.67%**	**58.65%**	**58.61%**	**6.78%**	33.73%	34.92%	36.84%
DeepViT	**1.83%**	26.72%	29.63%	33.42%	3.62%	41.92%	42.54%	43.56%
O-DeepViT	1.24 %	**29.15 %**	**31.72%**	**34.09%**	**4.86%**	**51.41%**	**51.64%**	**52.09%**
CaiT	48.21%	49.38%	49.52%	48.97%	**16.78%**	25.25%	25.37%	25.42%
O-CaiT	**56.43%**	**56.6%**	**55.78%**	**56.43%**	4.01%	**53.22%**	**54.01**%	**54.46%**

Table 4: Comparison between O-ViTs and ViTs with Noises

Method	YALE				ImageNet50			
	std = 1	std = 0.1	std = 0.08	std = 0.05	std = 1	std = 0.1	std = 0.08	std = 0.05
BaseViT	2.19%	65.15%	79.94%	94.88%	**44.76%**	**45.21%**	**45.28%**	**45.92%**
O-BaseViT	**4.75%**	**74.07%**	**85.38%**	**96.37%**	44.72%	45.02%	45.16%	45.64%
DeepViT	4.16%	64.30%	67.02%	70.92%	**4.16%**	30.72%	31.24%	31.68%
O-DeepViT	**5.76 %**	**96.16%**	**97.28%**	**99.04%**	3.72%	**34.08%**	**35.08%**	**34.72%**
CaiT	2.78%	3.63%	3.68%	3.74%	42.88%	43.0%	43.12%	43.28%
O-CaiT%	**6.30%**	**98.72%**	**99.04**%	**99.31%**	**46.20%**	**50.61%**	**50.60%**	**51.35%**