Rethinking 3D Convolution in ℓ_p-norm Space

Li Zhang[1,2,4], **Yan Zhong**[3], **Jianan Wang**[4], **Zhe Min**[5], **Rujing Wang**[1,2], **Liu Liu**[6*]
1 Hefei Institute of Physical Science, Chinese Academy of Sciences
2 University of Science and Technology of China, Hefei, China
3 School of Mathematical Sciences, Peking University. Beijing, China
4 Astribot, Shenzhen, China
5 Shandong University, Jinan, China
6 Hefei University of Technology, Hefei, China
zanly@mail.ustc.edu.cn, zhongyan@stu.pku.edu.cn

Abstract

Convolution is a fundamental operation in the 3D backbone. However, under certain conditions, the feature extraction ability of traditional convolution methods may be weakened. In this paper, we introduce a new convolution method based on ℓ_p-norm. For theoretical support, we prove the universal approximation theorem for ℓ_p-norm based convolution, and analyze the robustness and feasibility of ℓ_p-norms in 3D point cloud tasks. Concretely, ℓ_∞-norm based convolution is prone to feature loss. ℓ_2-norm based convolution is essentially a linear transformation of the traditional convolution. ℓ_1-norm based convolution is an economical and effective feature extractor. We propose customized optimization strategies to accelerate the training process of ℓ_1-norm based Nets and enhance the performance. Besides, a theoretical guarantee is given for the convergence by *regret* argument. We apply our methods to classic networks and conduct related experiments. Experimental results indicate that our approach exhibits competitive performance with traditional CNNs, with lower energy consumption and instruction latency.

1 Introduction

The convolution-based 3D backbone networks have demonstrated substantial success in foundational tasks such as classification [1], object tracking [2], scene segmentation [3], etc. Some downstream tasks also heavily rely on these networks, such as interactive perception [4], object manipulation [5], imitation learning [6], and human-machine collaboration [7]. In the traditional 3D convolution, suppose $K \in \mathbb{R}^{m\times n}$ is the filter, and $P_t \in \mathbb{R}^{m\times n}$ is the sampled matrix from the t-th sliding window on input data, $1 \le t \le T$. T is the total sliding counts. For any $t \ge 1$, the t-th convolution is calculated as:

$$P_t \odot K = \sum_{1\le i\le m} \sum_{1\le j\le n} P_t(i,j)\cdot K(i,j) \tag{1}$$

which is the same as inner product between vectors. To distinguish it from our new convolution framework, we refer to it as *inner product based convolution* in the following discussion. A geometric consideration arises when P_t follows a certain symmetric distribution, such as a Gaussian or uniform distribution. By symmetry, there exist some of $\{P_t\}_{t=1}^T$ situated close to the subspace perpendicular to K, which means $K \odot P_t \approx 0$. This inevitably leads to explicit feature loss, diminishing the model's ability on information extraction.

*Li Zhang and Yan Zhong contribute equally. This work was done when Li Zhang was an intern at Astribot. Corresponding author: Liu Liu.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

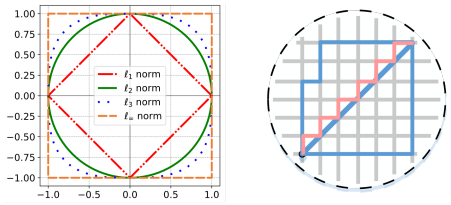
 https://doi.org/10.52202/079017-2889

In previous works, ℓ_p-norms ($p = 1, 2, 3, \cdots, \infty$) demonstrated strong performance across various domains [8, 9, 10]. These norms exhibit remarkable capabilities in expressing spatial structures and local relationships within sets of points. To address the limitations of inner product-based convolution in certain extreme cases and to explore the potential of ℓ_p-norms in feature extraction, we propose ℓ_p-norm-based convolution, *i.e.*, for any kernel K and sampled matrix P_t, it can be formulated as Eq. 2:

$$\|P_t - K\|_p \triangleq \Big(\sum_{1 \le i \le m} \sum_{1 \le j \le n} (P_t(i,j) - K(i,j))^p \Big)^{1/p}. \tag{2}$$

More precisely, the goal of this paper is to leverage the power of ℓ_p-norm measurement (Fig. 1 (a)) and devise efficient and robust optimization methods for it. Our solutions are as follows:

From the theoretical standpoint, we prove the universal approximation theorem of ℓ_p-norm Nets (for $p = 1, 2, 3, \cdots, \infty$). Besides, we show that ℓ_p-norm based convolutions are more robust than the traditional ones via variance analysis under random noise.



(a) ***(b)***

Figure 1: (**a**) Visualizing the circles of ℓ_p-norms. (**b**) Manhattan distance based ℓ_1-norm measure.

From the practical standpoint, we first discuss the performance of different ℓ_p-norms in actual execution. 3D convolution in ℓ_∞-norm space tends to lose multiple useful pieces of information since only the maximum absolute value is reserved. The ℓ_2-norm measure is inherently a linear transformation of the traditional convolution (details can be found in Sec. A). In contrast, the ℓ_1-norm has unique potential for 3D point cloud tasks. However, directly replacing traditional convolution with an ℓ_1-norm-based one is not feasible in practice due to the difficult convergence and local optima. To enhance network performance, we propose customized optimization strategies. The first strategy is a mixed gradient strategy (MGS), and the second is a dynamic learning rate controller (DLC). These strategies are applied in the training process (Algorithm 1) to accelerate network convergence and avoid local optima. We also provide a convergence guarantee for our optimization strategies from the perspective of *regret*.

We evaluate our method on several benchmarks, ranging from global, semi-dense, and dense prediction tasks. The experimental results show that ℓ_1-norm Net has the same competitive performance as traditional convolution. Moreover, the proposed ℓ_1-norm Net has three advantages: 1) ℓ_1-norm (inherently addition operation) has lower computational complexity compared to multiplication; 2) addition significantly reduces energy consumption [11]; 3) ℓ_1-norm operations (addition) has lower instruction latencies [12] than inner product process (multiplication). These properties facilitate the 3D point cloud tasks especially online tasks such as 3D real-time object detection, pose tracking, etc.

Contributions. 1) We prove the universal approximation for ℓ_p-norm Nets. And we show that ℓ_p-norm Nets are robust under random noise. 2) We compare different ℓ_p-norm based convolutions, and further propose a reliable and efficient ℓ_1-norm Net for 3D point cloud tasks with customized optimization strategies. We also give a theoretical guarantee for convergence by regret argument. 3) Experimental results demonstrate the effectiveness of our methods in 3D point cloud tasks, showing lower energy consumption and faster instruction execution.

2 Related Work

Different Convolution Methods. Convolutions have seen significant success, leading to various convolution methods aimed at improving performance and efficiency. Traditional convolutions, introduced by [13], use fixed-size kernels to extract features but are computationally intensive and may not capture diverse patterns effectively. To overcome these limitations, several alternatives have been proposed: 1) depthwise separable convolutions [14, 15]. Popularized by MobileNets, these decompose standard convolutions into depthwise and pointwise operations. 2) dilated convolutions [16, 17, 18]. These introduce spaces between kernel elements, expanding the receptive field without increasing parameters. 3) deformable convolutions [19, 20]. These adapt the sampling loca-

https://doi.org/10.52202/079017-2889

tions of the convolutional kernel, enhancing the network's ability to model geometric transformations. However, due to their unique strengths, they only excel at some specific tasks.

ℓ_p-norm Measure in Different Tasks. Using the ℓ_p-norm as a feature measurement function for convolutional kernels offers several advantages: 1) Flexibility: The ℓ_p-norm allows adjusting the parameter p according to specific needs [21, 22, 23]. 2) Sparsity: It encourages most elements in the convolutional kernel to approach zero, reducing computational complexity and storage requirements [21, 24]. Overall, in diverse settings, employ distinctive approaches. The ℓ_p-norm is widely used across various fields. For example, in *image processing*, the ℓ_1-norm is used for sparse representation in image compression [25], enabling efficient storage and transmission. In *machine learning and optimization*, optimization problems also use ℓ_p-norm constraints to impose sparsity or specific patterns in solutions [26, 27]. Despite progress, directly migrating these methods into 3D point cloud tasks causes a *domain gap*. In this work, we aim to explore ℓ_p-norm measure for 3D point cloud tasks in depth.

3 Methodology

Notations. For the sake of simplicity, in what follows, we take the classic PointNet++ [28] as the basis model to estimate the efficiency of ℓ_p-norm based Nets with the proposed optimization strategies. Note that, we directly replace the inner product based convolution by ℓ_p-norms ($p = 1, 2, 3, \ldots, \infty$) based one, and denote the corresponding network by ℓ_p-PointNet++ or ℓ_p-norm Net. Moreover, the proposed ℓ_p-norm based convolution can also be called ℓ_p-norm neuron.

3.1 Universal Approximation

The universal approximation ability of a neural network is crucial. Firstly, it establishes a solid theoretical foundation for the network's capabilities [29], which asserts that certain architectures and activation functions enable neural networks to approximate any continuous function. There is a series of works on the approximation capacity, such as theories for feedforward networks [30], RNNs [31], Transformer [32]. However, the universal approximation property of ℓ_p-PointNet++ has not been studied thoroughly up to now.

Theorem 1. *Assume $S = \{x_1, \cdots, x_N\} \subset \mathbb{R}^k$ is an arbitrary point cloud. $J \subset \mathbb{R}^k$ is any compact set and $S \subset J$. For any continuous function f defined on 2^J with respect to Hausdroff distance $d_H(\cdot, \cdot)$, there exists an ℓ_p-PointNet++ $\mathcal{P}$ satisfying for any $\epsilon > 0$,*

$$|f(S) - \mathcal{P}(S)| \leq \epsilon. \tag{3}$$

Moreover, for any ℓ_1-integrable function g defined on J, there exists an ℓ_p-PointNet++ $\mathcal{P}'$, for any $\epsilon' > 0$,

$$\int_{x \in J} |g(x) - \mathcal{P}'(x)| dx < \epsilon'. \tag{4}$$

Briefly speaking, f could be approximated by an MLP consisting of ℓ_p-norm convolution layers and a max pooling layer. And g could be approximated by a network composed of an ℓ_p-norm convolution layer and a fully connected layer. The detailed proof can be found in Sec. A from the appendix.

3.2 Robustness Analysis

In the following, we show that under Gaussian random noise on input data, ℓ_p-norm based convolutions are more robust than that based on inner product. Suppose $G \in \mathbb{R}^{m \times n}$ is a Gaussian matrix. Each $G(i, j) \sim N(0, \sigma^2)$ where $\sigma > 0$ is a constant. Let $P_t \in \mathbb{R}^{m \times n}$ be the data at time t and $K \in \mathbb{R}^{m \times n}$ be the kernel function.

For inner product,

$$Var\big[(G + P_t) \odot K\big] = \mathbb{E}_G\Big[\big(G \odot K - \mathbb{E}_G[G \odot K]\big)^2\Big] = Var\big[G \odot K\big], \tag{5}$$

and

$$G \odot K = \sum_{i=1}^{m} \sum_{j=1}^{n} G(i, j) K(i, j) \sim N\bigg(0, \sigma^2 \cdot \sum_{i=1}^{m} \sum_{j=1}^{n} K(i, j)^2\bigg). \tag{6}$$

https://doi.org/10.52202/079017-2889

Suppose $\forall i \in [m]$ and $\forall j \in [n]$, $K(i,j)$ is a constant, we have $Var\big[(G+P_t)\odot K\big] = \Theta(mn)$.

Table 1: **Variance of the ℓ_p-norm of Gaussian random vector when $mn = 9$.**

p	1	2	3	4	5
Var	3.24655	0.48327	0.31248	0.27494	0.26078
p	**6**	**7**	**8**	**9**	∞
Var	0.26093	0.26040	0.26078	0.26145	0.26875

For ℓ_p-norm, first we could prove that when $p=2$, $Var\big[\|G+X-K\|_2\big] = O(1)$, which is significantly smaller than $Var\big[(G+P_t)\odot K\big]$. The details of calculation could be found in Sec. A from the appendix. For the more general cases ($p = 1, 2, 3, \cdots, \infty$), we show that ℓ_p-norm has a small variance through numerical computation in the Tab. 1, where we take $\sigma = 1$.

3.3 Implementation of ℓ_p-norm Nets

Note that although Theorem 1 guarantees a universal approximation capability, it does not mean that all the ℓ_p-norm Nets are efficient and feasible in practice. Therefore, we further discuss the characteristics of each specific ℓ_p-norm Nets ($p = 1, 2, 3, \cdots, \infty$) in detail.

Assume the input data follows Gaussian distribution, saying G is the standard Gaussian matrix. For ℓ_p-norm based convolution, when p is greater than or equal to 3, the distribution of the output data is very close. We present the simulation results in Fig. 2. It's clear that when p is getting larger, the distribution of $\|G\|_p$ gradually overlaps with the distribution of $\|G\|_\infty$. Therefore, we take $p = \infty$ as the representative case for $p \geq 3$.

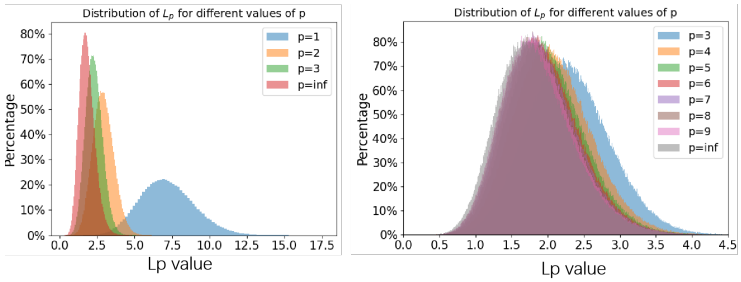


Figure 2: (**Left**) The distribution of $\|G\|_p$,where G is the standard Gaussian vector, $p = 1, 2, 3, \infty$ and $dim(G) = 9$. (**Right**) The distribution of $\|G\|_p$, $p = 3, 4, 5, 6, 7, 8, 9, \infty$ and $dim(G) = 9$.

Actually, l_∞-norm exhibits weaknesses due to its overly simplistic emphasis on the largest element. Namely this approach tends to oversimplify the feature space by disproportionately emphasizing only one dimension, potentially discarding valuable information present in other dimensions. Also, this concept is supported by experimental results in Sec. 5. Besides, ℓ_2-norm inherently is calculated by taking the square root of the sum of the squares of its elements. And ℓ_2-norm based convolution $\mathcal{C}_{\ell_2}$ can be regarded as an equivalence transformation of the traditional convolution $\mathcal{C}$. Briefly speaking, we could show that $\mathcal{C}_{\ell_2}^2 = \alpha + \beta \times \mathcal{C}$, where α and β are constants.

ℓ_1-norm can synthesize each element of the feature vector. And the ℓ_1-norm Net is not equivalent to a translation transform, which we believe holds potential as a 3D convolutional similarity metric function according to the Theorem. 1. To this end, our method focuses on rationalizing the ℓ_1-norm measure to maximize its potential in feature extraction. Mathematically, if the similarity measurement function between the input data and kernel function is replaced with the ℓ_1-norm, the convolution can be re-formulated as:

$$Y(P_t, K) = -\sum_{t\geq 1}\sum_{i,j} |P_t(i,j) - K(i,j)| \tag{7}$$

The underlying operation of ℓ_1-norm kernel function is addition, which has more development potential and application value in real scenarios. Specifically, 1) It contains almost no multiplication but addition, resulting in lower computational complexity of the model. 2) ℓ_1-norm operation (addition) is proved to have lower energy consumption compared to the inner product (multiplication) calculation [33]. Take the operation of floating-point addition and multiplication as an example, which has energy costs of 0.9 pJ and 3.7 pJ, respectively. 3) Low latency is also a consideration in practical application scenarios. [12] tells us that multiplication (inner product process) has longer theoretical instruction wait times than addition operations. Table 1 of this study lists the instruction

latency, throughput, and micromanipulation faults for Intel, AMD, and VIA CPUs. For instance, the latency of float multiplication and addition is 4 and 2 in the VIA Nano 2000 series.

3.4 Regret

It's a good way [34, 35, 36] to demonstrate the convergence of an optimization process by analyzing the *regret*. Performance measurement [37], optimization guidance [38], and feedback mechanisms [39] can be summarized as its advantages. We employ it the construct the convergence theorem for our optimization strategies in Sec. 4.

Consider a general online optimization model between a player and an adversary. A subset $\mathcal{F} \in \mathbb{R}^m$ is non-empty, bounded and closed. For each iteration $k \in [T^*]$, the player choose a point $\mathbf{x}_k \in \mathcal{F}$ (T^* is not known for player). After committing to this choice, a convex function h_k will be revealed by the adversary. And we note the cost of this game by *regret*:

$$R_{T^*} = \sum_{k=1}^{T^*} h_k(\mathbf{x}_k) - \min_{\mathbf{x}\in\mathcal{F}} \sum_{k=1}^{T^*} h_k(\mathbf{x}). \tag{8}$$

The player aims to carefully select $\mathbf{x}_k$ to minimize regret as much as possible, while conversely the adversary aims to specifically choose h_k to hinder the player. Intuitively, if an algorithm(the player) could bound regret by a sub-linear function of T^*, *i.e.*, $R_{T^*} = o(T^*)$, we could conclude that "on the average" the algorithm performs as well as the best fixed strategy in hindsight [40].

4 Optimization

By the argument above, we are motivated to devise a new convolution based on ℓ_1-norm. However, direct training of ℓ_1-norm Nets can easily lead to unsatisfactory results. Thus, two customized optimization strategies are proposed for training. Before introducing these optimization strategies, we clarify the notations in the following.

Notations Recall that $K \in \mathbb{R}^{m\times n}$ is the kernel and $P_t \in \mathbb{R}^{m\times n}$ is the sliding window on the input data, $1 \le t \le T$. $Y(P_t, K)$ is the convolution of K and P_t. L denotes the loss function in training process. We use the $m \times n$ matrix $\frac{\partial L}{\partial K}$ to denote the gradient on of L on K, where $(\frac{\partial L}{\partial K})_{i,j} = \frac{\partial L}{\partial K(i,j)}$. Besides, define the vectors

$$\frac{\partial L}{\partial Y} \triangleq \Big(\frac{\partial L}{\partial Y(P_1,K)}, \frac{\partial L}{\partial Y(P_2,K)}, \ldots, \frac{\partial L}{\partial Y(P_T,K)}\Big)$$

and

$$\frac{\partial Y}{\partial K(i,j)} \triangleq \Big(\frac{\partial Y(P_1,K)}{\partial K(i,j)}, \frac{\partial Y(P_2,K)}{\partial K(i,j)}, \ldots, \frac{\partial Y(P_T,K)}{\partial K(i,j)}\Big)$$

4.1 MGS: Mixed Gradient Strategy

Now we focus on the gradient descent in training process, especially the partial derivative of loss function L on the kernel K. It should be pointed out that L is a function on $(Y(P_1,K), Y(P_2,K), \ldots, Y(P_T,K))$. By chain rule of derivation we have for any given (i,j),

$$\frac{\partial L}{\partial K(i,j)} = \sum_{t=1}^{T} \frac{\partial L}{\partial Y(P_t,K)} \cdot \frac{\partial Y(P_t,K)}{\partial K(i,j)} = \Big\langle \frac{\partial L}{\partial Y}, \frac{\partial Y}{\partial K(i,j)} \Big\rangle \tag{9}$$

Notice that when loss function L is fixed, $\frac{\partial L}{\partial Y}$ is regardless of the choice of $Y(P_t,K)$ (inner product or ℓ_p-norm). And we should only focus on the vector $\frac{\partial Y}{\partial K(i,j)}$. In the context of ℓ_1-PointNet++:

$$\frac{\partial Y(P_t,K)}{\partial K(i,j)} = \mathrm{sgn}\big(P_t(i,j) - K(i,j)\big). \tag{10}$$

Here, $\mathrm{sgn}(\cdot)$ represents the sign function.

https://doi.org/10.52202/079017-2889

There are two unavoidable problems: 1) the use of Eq. 10 results in a signSGD update. As discussed in [41], the direction of signSGD is not aligned with the steepest descent, and this misalignment exacerbates with increasing dimensionality. 2) The gradient of ℓ_1-norm Net is significantly smaller than that of inner product convolution in the experiment. Namely, $\|\frac{\partial L}{\partial K}\|_2$ is extremely small when we choose the convolution Y as ℓ_1-norm. Taking PointNet++ on S3DIS as an example, we report the ℓ_2 norm of gradient of ℓ_1-PointNet++ in Fig. 3. The gradient from ℓ_1-PointNet++ is much smaller than that in PointNet++ (*e.g.*, ℓ_1-PointNet++: 0.0002, PointNet++: 0.3162 in layer I). Hence, this small gradient $\frac{\partial L}{\partial K}$ in ℓ_1-norm Net would significantly slow down the training process.

Based on the above observations, we introduce a novel **M**ixed **G**radient **S**trategy (**MGS**) tailored for ℓ_1-PointNet++ training. This approach strategically combines the gradients of the ℓ_1-PointNet++ and that of ℓ_2-PointNet++:

$$\frac{\partial Y(P_t, K)}{\partial K(i,j)} = \frac{P_t(i,j) - K(i,j)}{||K - P_t||_2}, \quad (11)$$

Actually, as we discussed above, ℓ_2-norm based convolution is a linear transform of inner product convolution. So gradient of ℓ_2-norm Net has a proper scale. The mixed strategy involves dynamically adjusting $\frac{\partial Y(P_t,K)}{\partial K(i,j)}$ during training, guided by a parameter $0 < \lambda < 1$ and the training step k. **The mixed gradient strategy** is expressed as:

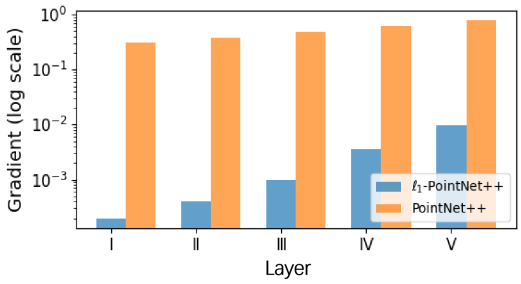


Figure 3: **The gradient of weight in each layer using two different networks at 1st iteration.** Layer I to III represent 3 SetAbstractions modules in ℓ_1-PointNet++ and layer IV to V represent fully connected layers. Note that the y-axis is on a logarithmic scale to reflect the magnitude of the values.

$$\frac{\partial Y(P_t, K)}{\partial K(i,j)} = (1 - \lambda^k)\text{sgn}(P_t(i,j) - K(i,j)) + \lambda^k(P_t(i,j) - K(i,j)). \quad (12)$$

This dynamic adjustment introduces a controlled transition in the gradient computation as training progresses. Taking $\lambda = 0.99$ for example, when k is small, the term λ^k dominates and $\frac{\partial Y(P_t,K)}{\partial K(i,j)}$approximates to $P_t(i,j) - K(i,j)$. This initial configuration aligns with the more efficient ℓ_2-like update, providing stability and aiding in faster convergence. As training progresses (k gets larger), the term λ^k becomes more prominent, shifting the gradient computation towards $\text{sgn}(P_t(i,j) - K(i,j))$. This transition allows the model to leverage the advantages of the ℓ_1-PointNet++ structure, facilitating sparsity in the learned features. By dynamically adapting the gradient computation based on the training step, the mixed strategy offers a flexible and adaptive approach to overcome the challenges associated with fixed gradient schemes. This dynamic adjustment provides a thoughtful compromise, combining the efficiency of ℓ_2-like updates in the initial stages with the sparsity-inducing benefits of ℓ_1-PointNet++ in later stages.

In fact, there is quite a bit of literature supporting the effectiveness of the signSGD update scheme, and in particular, it has been shown that it has some advantages in avoiding saddle points [42]. However, when certain random rotations of the objective appear, signSGD may become trapped in a periodic behavior that hinders convergence in such cases. To address this unexpected behavior, we additionally explored the introduction of momentum into the update rule. Our experimental results prove that this modification effectively breaks the symmetry induced by random rotations, preventing the model from getting stuck and fostering smoother convergence.

4.2 DLC: Dynamic Learning rate Controller

Considering the uniqueness of the mixed gradient strategy, we focus on achieving larger update magnitudes and faster convergence rates during the initial stages of training. However, in the later stages, we aim to revert to signSGD, implementing a more cautious update strategy to enhance the model's precision. Therefore, we propose a learning rate update strategy that adapts to this characteristic: **D**ynamic **L**earning rate **C**ontroller (**DLC**), maintaining a higher rate in the early training phase, and returning to a lower rate in the later phase.

To this end, we design two bound functions to control the learning rate: the lower bound

$$\alpha_1(k) = p_1 \cdot (1 + \frac{p_2}{e^k}) \tag{13}$$

and the upper bound

$$\alpha_2(k) = p_1 \cdot (1 + \frac{p_3}{k}) \tag{14}$$

where p_1, p_2 and p_3 are hyper-parameters to be determined and k denotes the training step. And we use simple comparison operations to make learning rate $\alpha(k)$ locate in $[\alpha_1(k), \alpha_2(k)]$:

$$\hat{\alpha}(k) \leftarrow \min\big\{\max\{\alpha_1(k), \mathcal{A}[\alpha(k)]\}, \alpha_2(k)\big\}. \tag{15}$$

To enhance the universality of this dynamic control framework, $\mathcal{A}$ could be another learning rate optimization algorithm like the adaptive learning rate strategy of [43], which can be specifically switched according to the task at hand. However, regardless of $\mathcal{A}$, we will later demonstrate that dynamic control alone is sufficient to provide theoretical convergence guarantees by the regret argument of Theorem 2, and it also performs well in experiments.

4.3 Training Framework

It has been noted from previous discussions that the momentum method can help signSGD avoid getting trapped in cycles, thereby improving training stability. Combining the methods above, we present the global optimization algorithm (**O**ptimizer with **M**ixed gradient strategy and **D**ynamic learning rate controller, **OMD**) for ℓ_1-PointNet++ training. Details are shown in Algorithm. 1

Algorithm 1 OMD

Input: Initial learning rate α, hyper-parameters p_1, p_2, p_3, referred by Eq. 13 and Eq. 14. q_0 and q in $(0, 1)$.

$\mathbf{m}_0 = 0$, $\alpha(0) = \alpha$, $\mathbf{x}_1 = \vec{0}$.
Set the functions $\alpha_1(k)$ and $\alpha_2(k)$ by hyper-parameters p_1, p_2, p_3.
for $k = 1$ to T^* **do**
  $\mathbf{g}_k \leftarrow \frac{\partial L}{\partial K}(\mathbf{x}_k)$ # Consider the gradient $\frac{\partial L}{\partial K}$ as an vector here. $\frac{\partial L}{\partial K(i,j)} = \langle \frac{\partial L}{\partial Y}, \frac{\partial Y}{\partial K(i,j)} \rangle$. $\frac{\partial L}{\partial Y}$ only depends on the choice of loss function. See Eq. 12 for $\frac{\partial Y}{\partial K(i,j)}$.
  $q_k = q_0 \cdot q^k$.
  $\mathbf{m}_k = q_k \cdot \mathbf{m}_{k-1} + (1 - q_k) \cdot \mathbf{g}_k$
  $\hat{\alpha}(k) \leftarrow \min\big\{\max\{\alpha_1(k), \mathcal{A}[\alpha(k)]\}, \alpha_2(k)\big\}$
  $\alpha(k) \leftarrow \hat{\alpha}(k)/\sqrt{k}$
  $\mathbf{x}_{k+1} = \Pi_{\mathcal{F}, \alpha(k)^{-1/2}} \left(\mathbf{x}_k - \alpha(k) \cdot \mathbf{m}_k\right)$
end for

Here we give a convergence guarantee for **OMD** under an online optimization framework, which is harder than offline optimization. We could show that regret R_{T^*} of **OMD** is bounded by $O(\sqrt{T^*})$. Low regret means the algorithm progressively gets closer to the optimal solution over time. This shows that **OMD** has reliable convergence properties, making it a dependable optimization method.

Theorem 2. *Continue with the settings and notations of Algorithm 1. Suppose $\mathcal{F} \subset \mathbb{R}^n$ is bounded, saying $max_{\mathbf{x},\mathbf{y}\in\mathcal{F}} \|\mathbf{x} - \mathbf{y}\|_\infty \leq B_\infty$ Besides, suppose $\forall k \in [T^*]$, $\|\mathbf{g}_k\|_2 \leq B_2$. we could show that for any convex functions $\{h_k\}_{t=1}^{T^*}$,*

$$R_{T^*} = \sum_{k=1}^{T^*} h_k(\mathbf{x}_k) - \sum_{t=1}^{T^*} h_k(\mathbf{x}^\star) \leq C_1 \cdot \sqrt{T^*} + C_2$$

where C_1 and C_2 are constants that rely on p_1, p_2, p_3, B_∞, B_2, q_0 and q. And $\mathbf{x}^\star \triangleq arg\min_{\mathbf{x}\in\mathcal{F}} \sum_{k=1}^{T^*} h_k(\mathbf{x})$.

The proof could be found in the appendix, Sec. A.

5 Experiments

To validate the generalizability and robustness of the method and thus ensure its effectiveness and broad applicability, we verify the performance of our method in several tasks, ranging from **Global Tasks** (*i.e.*, Parts Segmentation), **Semi-dense Prediction** (*i.e.*, scenario semantic segmentation), and **Dense Prediction** (*i.e.*, pose estimation) tasks. Shapenet, S3DIS, and GarmentNets Simulation are used as the datasets.

https://doi.org/10.52202/079017-2889

5.1 Dataset and Experimental Settings

Dataset. 1) ShapeNet. In ShapeNet, there are 16,881 shapes from 16 categories, which are annotated with 50 parts in total. Note that most object categories are labeled with two to five parts and Ground Truth annotations are labeled on sampled points on the shapes. This task can be regarded as a point-wise classification task. 2) S3DIS. The Stanford Large-Scale 3D Indoor Spaces Dataset, which encompasses 3D scans obtained from Matterport scanners across 6 distinct areas, comprising a total of 271 rooms. Within the S3DIS dataset, every point within the scans is labeled with a semantic category from a set of 13 distinct classes. These classes encompass various elements such as chairs, tables, floors, walls, among others, in addition to a category for clutter. 3) GarmentNets Simulation. GarmentNets Simulation is a large-scale dataset proposed by [44]. This dataset has six garment categories with a total data volume of 1.72TB. Dress, Jump, Skirt, Top, Pants and Shirt are included.

Experimental Settings. We train our frameworks using CrossEntropy loss and the AdamW optimizer [45], with an initial learning rate of 0.001, a weight decay of 10^{-4}, Cosine Decay, and a batch size of 32. The total training consists of 200 epochs. *All tasks use the same settings unless otherwise specified.* All experiments are conducted on a computer workstation with three GeForce GTX 3090 GPUs using the PyTorch deep learning framework. The best model on the validation set is selected for testing.

5.2 Experiments on Global Task

Parts Segmentation. As a classic global task, 3D object parts segmentation is an important predecessor for articulated objects from the embodied intelligence community, such as pose estimation [46, 47], manipulation [48, 49], etc. In this section, we conduct experiments on ShapeNet part dataset [50].

Table 2: **Quantitative segmentation results on ShapeNet part dataset**. Note that a 3D fully convolutional network is proposed as the 3DCNN, mIoU(%) is reported as the metric on points.

Model	Mean	Shape Names															
		aero	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skate board	table
# shape number		2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
3DCNN	79.4	75.1	72.8	73.3	70.0	87.2	63.5	88.4	79.6	74.4	93.9	58.7	91.8	76.4	51.2	65.3	77.1
ℓ_1-3DCNN	**79.4**	79.3	70.9	71.3	72.9	86.3	58.6	90.0	76.5	74.9	92.6	63.8	89.9	75.8	55.8	63.6	79.2
PointNet	**83.7**	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
ℓ_1-PointNet	83.3	85.9	76.5	78.3	78.3	86.1	75.6	89.8	85.3	81.0	97.7	62.3	91.2	83.9	60.1	73.0	80.7
PointNet++	86.2	87.1	80.3	86.3	74.3	90.1	75.3	92.9	86.3	79.3	94.9	71.2	93.3	81.9	59.2	73.8	81.2
ℓ_1-PointNet++	**86.5**	89.1	78.3	85.2	76.3	86.9	74.6	93.1	84.9	79.6	94.6	72.6	91.3	81.3	66.3	74.8	81.8

From Tab. 2, see them all: we find that our method has almost equivalent performance to the conventional method when being equipped with PointNet, and achieves superior performance on 3DCNN and PointNet++. Treat them equally: we see that our method can often perform better in some categories (*e.g.*, car, motor, rocket, etc.), these categories usually have a larger volume (*i.e.*, a more sparsified point cloud) compared to other objects. We propose that the inner product within convolutional networks has a tendency to highlight local context among points, yet it is greatly affected by the overall translation and scaling of the dataset. Our method focuses on the points drawn from ℓ_1-norm space and addresses this problem by integrating the inherent distance measure into our architecture. Specifically speaking, The Manhattan distance based ℓ_1-norm Nets tend to avoid this problem, which notices point cloud features at a longer distance.

5.3 Experiments on Semi-dense Prediction Task

Scenario Semantic Segmentation. As a semi-dense prediction task, this task aims to segment distinct regions within a 3D scene based on their semantic meaning using point cloud data. Semantic scene segmentation is crucial for understanding and interpreting the spatial arrangement and relationships between objects in 3D scenes. For our study, we utilize the S3DIS dataset. The metrics and experimental settings follow those outlined in [28].

Following the training and test strategies used in [51], we first divide the point cloud using the room as the basic unit and then sample the room at a size of 1m * 1m (randomly sampling up to 4096

Table 3: **(Left) Results on Semantic Segmentation in Scenes.** Metric is average IoU (%) over 13 classes (structural and furniture elements plus clutter) and classification accuracy is calculated on points. Our methods achieved competitive performance with significant energy reductions (**61%**).

Model	Mean IoU (%)	Overall Accuracy (%)	Energy (μj)
PointNet	**47.7**	**78.6**	7.981
ℓ_1-PointNet	47.6	77.9	**3.471**
PointNet++	53.5	**83.0**	3.395
ℓ_1-PointNet++	**53.9**	82.9	**1.328**

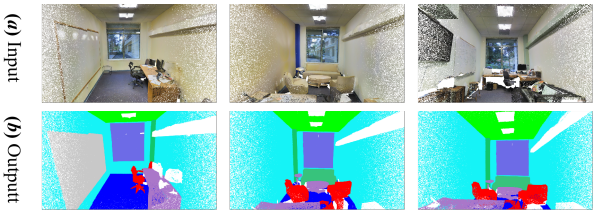


Figure 4: **(Right) Qualitative Results for Semantic Segmentation.** We put the colored point cloud on the top part (Input data) and put semantic segmentation results from the same camera viewpoint on points (Output) in the bottom part.

points during training, and all points are involved in the computation during testing), which in turn predicts the class of each point in each block. Note that we use a 9-dimensional vector to represent each point, representing XYZ, RGB, and normalized room location (ranging from 0 to 1). K-fold strategy is also used for training and testing.

The quantitative results are reported in Tab. 3. Experimental results show that although our approach achieves almost equivalent performance to inner product based networks, we maximize the potential of ℓ_1-norm measure by relying on our proposed optimization strategy, which allows us to achieve similar performance but with less computational complexity and lower energy consumption (Almost **61%** energy reductions). Also, we provide qualitative segmentation results for visualization in Fig. 4. Overall, our model generates consistent object predictions and is resilient to the presence of absent points and obstructions.

5.4 Experiments of Dense Prediction Task

Garment Pose Estimation Garments, vital in daily life, present unique challenges for machine perception and interaction due to properties like infinite degrees of freedom and thin structure. Garment pose estimation and tracking systems hold potential for applications in mixed reality [52, 53], augmented reality [54, 52], and robotic manipulation [55, 49]. Addressing these challenges, mainstream methods typically employ Normalized Object Coordinate Space (NOCS) [56] for **dense prediction tasks**. In this section, we introduce GarmentNets [44], a baseline focusing on garment pose estimation using partial point clouds as input and generating complete point clouds as output. Our approach utilizes the GarmentNets Simulation Dataset to evaluate this task. The total epoch number is 200, and the batch size is 16.

Table 4: **Quantitative Results on Garment Pose Estimation.** The metric is measured using Chamfer distance (cm) under the canonical pose. The lower is the better result.

Model	Dress	Jumpsuit	Skirt	Top	Pants	Shirt
GarmentNets [44]	1.94	**1.45**	2.00	1.30	1.03	1.70
ℓ_1-GarmentNets	**1.83**	1.56	**1.91**	**1.26**	**0.99**	**1.62**

Quantitative results are in Tab. 4. Note that we use Symmetric Chamfer Distance as the metric, This metric measures accuracy and completeness for surface reconstruction. The accuracy metric is defined as the mean L2 distance of points on the output mesh to their nearest neighbors on the GT mesh. From the table, it can be seen that our method performs comparably to the original method.

5.5 Ablation Experiments

Replacing Means. The most critical structure of PointNet++ is the 3 separate SetAbstractions modules (SA). Hence, to explore the effect of using the ℓ_1-PointNet++ at different places and in different ratios, we remove the modules at different ratios and places on S3DIS. The experimental result is shown in Tab. 5. In many aspects, we can infer that the average mean IOU and accuracy are higher under the 66.7% ratio than those reported under the 33.3% ratio. This result tells the conclusion that our ℓ_1-norm measure can exact more useful features from sparse point clouds. we hope these results can prompt further study on replacing means, such as different replacing ratios in

https://doi.org/10.52202/079017-2889

each inner module, creditable ways to combine hybrid convolutional blocks. etc. We leave this for more passionate researchers in the future.

Table 5: **Comparisons of Results on S3DIS with Different Replacing Ratio and Places**. We estimate the energy costs according to [11], *i.e.*, one operation of floating-point addition and multiplication have energy costs of 0.9 pJ and 3.7 pJ, respectively. SA: SetAbstractions module. ★ means that 33.3% replacing ratio of PointNet++ has #Add-0.492 M, #Mul-0.984 M, Energy-4.0836 μJ, while ♣ means that 66.7% replacing ratio of PointNet++ has #Add-0.984 M, #Mul-0.492 M, Energy-2.7060 μJ.

Replacing Ratio	ℓ_1-norm neuron? SA1	SA2	SA3	Mean IOU	Accuracy	Info
33.3%	✓			51.9%	79.8%	
		✓		52.3%	81.8%	★
			✓	52.5%	81.1%	
66.7%		✓	✓	53.2%	82.4%	
	✓	✓		53.0%	81.0%	♣
	✓	✓		52.2%	81.5%	

Table 6: **Ablation Results on S3DIS Dataset Using Different Variants of ℓ_1-Nets.** Mean IOU and overall Accuracy (%) are reported. Note that the results of ℓ_1-PointNet are reported from I to IV, and ℓ_1-PointNet++ are reported from V to VIII. Besides, vanilla Net represents the model without our customized optimization strategy while training.

Index	Optimization? MGS	DLC	Mean IOU (%)	Overall Accuracy (%)
I (Vanilla)			33.2%	56.3%
II	✓		39.6%	68.6%
III		✓	42.8%	70.1%
IV (Ours)	✓	✓	47.6%	77.9%
V (Vanilla)			38.9%	55.6%
VI	✓		43.6%	69.3%
VII		✓	48.4%	75.6%
VIII (Ours)	✓	✓	53.9%	82.9%

Optimization Strategy. As demonstrated in Sec. 4, we propose mixed gradient strategy (MGS) to accelerate network convergence, while dynamic learning rate controller (DLC) helps our network move away from local optima. To evaluate the effectiveness of MGS and DLC, we remove them separately from ℓ_1-PointNet++ and evaluate the scenario semantic segmentation performance on S3DIS. Tab. 6 presents the quantitative results. The baselines (I and V) indicate that we only use ℓ_1-norm as the similarity measurement but without any optimization. It can be observed that they both resulted in huge performance degradation. Besides, we can see that both our MGS and DLC contribute to network convergence and optimization results.

6 Limitations and Broader Impact

Firstly, some of the other convolutions (*e.g.*, sparse convolution, group convolution, dilated convolution) and additional computer vision tasks remain unexplored. Secondly, the inference speed of the ℓ_1-norm Net is marginally slower than that of traditional one. This is attributed to the lack of CUDA and cuDNN optimized operations for Manhattan distance metrics. It's noteworthy that, beyond introducing a novel convolution based on the ℓ_p-norm and proving the universal approximation theorem for theoretical support, this paper also presents customized optimization strategies.

7 Conclusion

In this paper, we are motivated to explore ℓ_p-norm measure to replace the classic inner product convolution. we first prove the universal approximation of ℓ_p-norm Nets. And then we compare different ℓ_p-norm measures and propose the ℓ_1-norm Net for 3D point cloud tasks. Furthermore, we design the customized optimization strategies (*i.e.*, mixed gradient strategy and dynamic control on learning rate) for ℓ_1-norm Net. When introducing our method to classical 3D networks, they achieve competitive performances at a lower energy cost. In summary, our ℓ_1-norm Net can achieve similar performance to traditional convolution network, but with less computational cost and lower instruction latency.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China under Grant 62302143 and Anhui Provincial Natural Science Foundation under Grant 2308085QF207. Thanks for the help of Xinyuan Song.

References

[1] Min Zhang, Yifan Wang, Pranav Kadam, Shan Liu, and C-C Jay Kuo. Pointhop++: A lightweight learning model on point sets for 3d classification. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 3319–3323. IEEE, 2020.

[2] Haozhe Qi, Chen Feng, Zhiguo Cao, Feng Zhao, and Yang Xiao. P2b: Point-to-box network for 3d object tracking in point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6329–6338, 2020.

[3] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 452–468, 2018.

[4] Qiaojun Yu, Junbo Wang, Wenhai Liu, Ce Hao, Liu Liu, Lin Shao, Weiming Wang, and Cewu Lu. Gamma: Generalizable articulation modeling and manipulation for articulated objects. *arXiv preprint arXiv:2309.16264*, 2023.

[5] Haoyu Xiong, Haoyuan Fu, Jieyi Zhang, Chen Bao, Qiang Zhang, Yongxi Huang, Wenqiang Xu, Animesh Garg, and Cewu Lu. Robotube: Learning household manipulation from human videos with simulated twin environments. In *Conference on Robot Learning*, pages 1–10. PMLR, 2023.

[6] Fan Zhang and Yiannis Demiris. Learning garment manipulation policies toward robot-assisted dressing. *Science robotics*, 7(65):eabm6010, 2022.

[7] Kailin Li, Lixin Yang, Haoyu Zhen, Zenan Lin, Xinyu Zhan, Licheng Zhong, Jian Xu, Kejian Wu, and Cewu Lu. Chord: Category-level hand-held object reconstruction via shape deformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9444–9454, 2023.

[8] Marius Kloft, Ulf Brefeld, Sören Sonnenburg, and Alexander Zien. Lp-norm multiple kernel learning. *The Journal of Machine Learning Research*, 12:953–997, 2011.

[9] Qingsong Gu and Po-Lam Yung. A new formula for the lp norm. *Journal of Functional Analysis*, 281(4):109075, 2021.

[10] James A Cadzow. Minimum ℓ_1, ℓ_2, and ℓ_∞ norm approximate solutions to an overdetermined system of linear equations. *Digital Signal Processing*, 12(4):524–560, 2002.

[11] William Dally. High-performance hardware for machine learning. *Nips Tutorial*, 2:3, 2015.

[12] Instruction latencies of different operations. `www.agner.org/optimize/instruction_tables.pdf`.

[13] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[14] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[15] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[16] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[17] Dren Gashi, Mike Pereira, and Valeriia Vterkovska. Multi-scale context aggregation by dilated convolutions machine learning-project. 2017.

[18] Xin Wang, Rongrong Lv, Yang Zhao, Tangwen Yang, and Qiuqi Ruan. Multi-scale context aggregation network with attention-guided for crowd counting. In *2020 15th IEEE International Conference on Signal Processing (ICSP)*, volume 1, pages 240–245. IEEE, 2020.

 https://doi.org/10.52202/079017-2889

[19] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[20] Feng Chen, Fei Wu, Jing Xu, Guangwei Gao, Qi Ge, and Xiao-Yuan Jing. Adaptive deformable convolutional network. *Neurocomputing*, 453:853–864, 2021.

[21] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005.

[22] Julianne Chung and Silvia Gazzola. Flexible krylov methods for $\backslash$ ell_p regularization. *SIAM Journal on Scientific Computing*, 41(5):S149–S171, 2019.

[23] Martine Labbé, Justo Puerto, and Moisés Rodríguez-Madrena. Shortest paths and location problems in a continuous framework with different $\backslash$ ell_p-norms on different regions. *arXiv preprint arXiv:2110.07866*, 2021.

[24] Jinglai Shen and Seyedahmad Mousavi. Least sparsity of p-norm based optimization problems with p>1. *SIAM Journal on Optimization*, 28(3):2721–2751, 2018.

[25] CS Sastry and Ashish Mishra. Application of l1-norm minimization technique to image retrieval. *World Academy of Science, Engineering and Technology*, 56(145):801–804, 2009.

[26] Twan Van Laarhoven. L2 regularization versus batch and weight normalization. *arXiv preprint arXiv:1706.05350*, 2017.

[27] Stephen Boyd and Venkataramanan Balakrishnan. A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its ℓ_∞-norm. *Systems & Control Letters*, 15(1):1–7, 1990.

[28] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[29] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[30] Non-Polynomial Activation Functions Can. Multilayer feedforward networks with non-polynomial activation functions can approximate any function–_-. 1991.

[31] Anton Maximilian Schäfer and Hans Georg Zimmermann. Recurrent neural networks are universal approximators. In *Artificial Neural Networks–ICANN 2006: 16th International Conference, Athens, Greece, September 10-14, 2006. Proceedings, Part I 16*, pages 632–640. Springer, 2006.

[32] Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank J Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*, 2019.

[33] Mark Horowitz. 1.1 computing's energy problem (and what we can do about it). In *2014 IEEE international solid-state circuits conference digest of technical papers (ISSCC)*, pages 10–14. IEEE, 2014.

[34] Noam Brown and Tuomas Sandholm. Regret transfer and parameter optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

[35] Yingjie Fei, Zhuoran Yang, Zhaoran Wang, and Qiaomin Xie. Dynamic regret of policy optimization in non-stationary environments. *Advances in Neural Information Processing Systems*, 33:6743–6754, 2020.

[36] Zi Wang, Beomjoon Kim, and Leslie P Kaelbling. Regret bounds for meta bayesian optimization with an unknown gaussian process prior. *Advances in Neural Information Processing Systems*, 31, 2018.

[37] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[38] Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

[39] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

[40] Elad Hazan. 10 the convex optimization approach to regret minimization. *Optimization for machine learning*, page 287, 2012.

[41] Jeremy Bernstein, Kamyar Azizzadenesheli, Yu-Xiang Wang, and Anima Anandkumar. Convergence rate of sign stochastic gradient descent for non-convex functions. 2018.

[42] Michael Crawshaw, Mingrui Liu, Francesco Orabona, Wei Zhang, and Zhenxun Zhuang. Robustness to unbounded smoothness of generalized signsgd. *Advances in Neural Information Processing Systems*, 35:9955–9968, 2022.

[43] Hanting Chen, Yunhe Wang, Chunjing Xu, Boxin Shi, Chao Xu, Qi Tian, and Chang Xu. Addernet: Do we really need multiplications in deep learning? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1468–1477, 2020.

[44] Cheng Chi and Shuran Song. Garmentnets: Category-level pose estimation for garments via canonical space shape completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3324–3333, 2021.

[45] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. 2019.

[46] Liu Liu, Han Xue, Wenqiang Xu, Haoyuan Fu, and Cewu Lu. Toward real-world category-level articulation pose estimation. *IEEE Transactions on Image Processing*, 31:1072–1083, 2022.

[47] Xiaolong Li, He Wang, Li Yi, Leonidas J Guibas, A Lynn Abbott, and Shuran Song. Category-level articulated object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3706–3715, 2020.

[48] Jianren Wang, Sudeep Dasari, Mohan Kumar Srirama, Shubham Tulsiani, and Abhinav Gupta. Manipulate by seeing: Creating manipulation controllers from pre-trained representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3859–3868, 2023.

[49] Yezhou Yang, Yi Li, Cornelia Fermuller, and Yiannis Aloimonos. Robot learning manipulation action plans by" watching" unconstrained videos from the world wide web. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29, 2015.

[50] Li Yi, Vladimir G Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (ToG)*, 35(6):1–12, 2016.

[51] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[52] Julie Carmigniani and Borko Furht. Augmented reality: an overview. *Handbook of augmented reality*, pages 3–46, 2011.

[53] Dhiraj Amin and Sharvari Govilkar. Comparative study of augmented reality sdks. *International Journal on Computational Science & Applications*, 5(1):11–26, 2015.

[54] Ronald T Azuma. A survey of augmented reality. *Presence: teleoperators & virtual environments*, 6(4):355–385, 1997.

 https://doi.org/10.52202/079017-2889

[55] Alper Canberk, Cheng Chi, Huy Ha, Benjamin Burchfiel, Eric Cousineau, Siyuan Feng, and Shuran Song. Cloth funnels: Canonicalized-alignment for multi-purpose garment manipulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5872–5879. IEEE, 2023.

[56] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019.

[57] Jooyoung Park and Irwin W Sandberg. Universal approximation using radial-basis-function networks. *Neural computation*, 3(2):246–257, 1991.

[58] H Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*, 2010.

– Appendix –

In the Appendix, we present additional information on our methods. Concretely, we provides a detailed theoretical analysis of the theorems from the main paper, including the variance analysis, the proposed universal approximation, the regret argument, and the equivalence of ℓ_2-norm Measure.

A Additional Theoretical Analysis

A.1 Omitted proof of variance analysis

Since adding a constant does not significantly affect variance, for ease of demonstration, we could assume $Var\big[\|G + P_t - K\|_2\big] \approx Var\big[\|G\|_2\big]$. Notice that

$$Var\big[\|G\|_2\big] = \mathbb{E}_{G\sim N(0,\mathbb{I}_m)}\big[\|G\|_2^2\big] - \Big(\mathbb{E}_{G\sim N(0,\mathbb{I}_m)}\big[\|G\|_2\big]\Big)^2. \tag{16}$$

It's easy to verify that for any $u \geq 0$,

$$\sqrt{u} \geq (1 + u - (u-1)^2)/2.$$

Let $u = \frac{\|G\|_2^2}{m}$ and calculate the expectation of G on both sides of the inequality, we have

$$\frac{\mathbb{E}[\|G\|_2]}{\sqrt{m}} \geq \frac{1}{2} \cdot \big(2 - \mathbb{E}\Big[(\frac{\|G\|_2^2}{m} - 1)^2\Big]\big). \tag{17}$$

Because $\mathbb{E}\big[(\frac{\|G\|_2^2}{m} - 1)^2\big] = \frac{1}{m^2} \cdot \mathbb{E}\big[\sum_{i=1}^m (G(i)^2 - 1)^2 + \sum_{i\neq j}(G(i)^2 - 1)(G(j)^2 - 1)\big]$ and $\forall i$, $\mathbb{E}[G(i)^2 - 1] = 0$, we can conclude that

$$\mathbb{E}\big[(\frac{\|G\|_2^2}{m} - 1)^2\big] = \frac{1}{m} \cdot \mathbb{E}[G(1)^4 + 1 - 2 \cdot G(1)^2] \tag{18a}$$

$$= \frac{2}{m}. \tag{18b}$$

where the first equation holds for all the $G(i)$s are i.i.d. The second equation holds for $\mathbb{E}[G(1)^4] = 3$, $\mathbb{E}[G(1)^2] = 1$. Combining inequality (17) and Equation (18b), $\mathbb{E}_{G\sim\mathcal{N}(0,\mathbb{I}_m)}\left[\|G\|_2\right] \geq \frac{\sqrt{m}}{2} \cdot \big(2 - \frac{2}{m}\big)$. Therefore, by Equation (16) we have

$$Var[\|G\|_2] < 2 - \frac{1}{m} = O(1).$$

Thus we have shown that $Var\big[\|(G + P_t) - K\|_2\big] = O(1)$.

A.2 Proof of Theorem 1

Scaling S and J by $\frac{1}{diam(J)}$ where $diam(J) = \max_{x,y\in J}\{\|x - y\|_\infty\}$, we could assume $S = \{x_1, \cdots, x_N\}$ and $\forall i \in [N]$ $x_i \in J \subset [0,1]^k$. For convenience, first we show the case $k = 1$. Here we refer the construction of soft occupancy function in [51]. Because f is continuous function, for any $\epsilon > 0$, $\exists\ \sigma > 0$ so that $|f(S_1) - f(S_2)| < \epsilon$ for any S_1 and S_2 with $d_H(S_1, S_2) < \delta$. Let $M = \lceil \frac{1}{\delta} \rceil$ and $h_m(x) = exp(-d_H(x, [\frac{m-1}{M}, \frac{m}{M}]))$ be the soft occupancy function, for all $m \in [M]$. Next, for all $m \in [M]$ define

$$\hat{v}_m(S) = \max_{x\in S}\{h_m(x)\} \tag{19}$$

and,

$$v_m(S) = \begin{cases} 1, & \hat{v}_m(S) \geq 1 \\ 0, & \hat{v}_m(S) < 1. \end{cases} \tag{20}$$

https://doi.org/10.52202/079017-2889

$v_m(S)$ indicates the occupancy of the m-th interval by points in S. Define $\mathbf{v} : 2^J \to \{0,1\}^M$ and for any $S \in 2^J$, $\mathbf{v}(S) = (v_1(S), v_2(S), \cdots, v_M(S))$. And then define $\eta : \{0,1\}^M \to 2^J$, $\eta(\mathbf{v}(S)) = \{\frac{m-1}{M} \mid v_m(S) \geq 1\}$. Notice that by this construction, $d_H(\eta(\mathbf{v}(S)), S) < \frac{1}{M} \leq \delta$. So let $\omega : \{0,1\}^M \to \mathbb{R}$ and $\omega(\mathbf{v}) = f(\eta(\mathbf{v}))$, we have

$$|\omega(\mathbf{v}(S)) - f(S)| = |f(\eta(\mathbf{v}(S))) - f(S)| < \epsilon \tag{21}$$

The last inequality holds for the definition of Hausdorff distance and continuity of f. Here ω and $\{h_m\}_{m=1}^M$ could be made up of a multi-layer perceptron network [51]. $\{\hat{v}_m\}_{m=1}^M$ consist of a max pooling layer on $\{h_m\}_{m=1}^M$ and $\{v_m\}_{m=1}^M$ can be composed of a simple perceptron layer on $\{\hat{v}_m\}_{m=1}^M$, which compares $\hat{v}_m(S)$ and 1. For the general cases $k \geq 1$, it suffices to get the same conclusion by simply extending the 1 dimensional functions h_m, $\hat{v}_m$, v_m to k dimension. So there is a ℓ_p-PointNet++ $\mathcal{P}$ that can approximate any continuous function f on 2^J.

We employ the RBF theory of [57] to give the second conclusion. For completeness, we restate it here.

Theorem 3 ([57]). *The radio basis networks consist of a family of functions(RBF) noted by S_K:*

$$\sum_{i=1}^{H} a_i \cdot K(\frac{x - z_i}{\sigma})$$

where $x \in \mathbb{R}^d$, $z_i \in \mathbb{R}^d$,$\sigma \in \mathbb{R}$, $H \in \mathcal{N}$. S_K is dense in $\ell_1(\mathbb{R}^d)$, if K satisfies: 1.integrable bounded, 2.K is continuous almost everywhere, 3.$\int K(x)dx \neq 0$.

It's clear that the ℓ_p-norm $\|\cdot\|_p : \mathbb{R}^d \to \mathbb{R}$ satisfies all the three conditions on K. Besides, a large enough ℓ_p based convolution layer with a full connected layer could represent all the functions $\sum_{i=1}^{H} a_i \cdot \|(\frac{x-z_i}{\sigma})\|_p$. So for any ℓ_1-integrable function g, there exists an ℓ_p-PointNet++ $\mathcal{P}'$ such that for any $\epsilon > 0$, $\int |g(x) - \mathcal{P}'(x)|dx < \epsilon$.

A.3 Proof of Theorem 2

Before proving, we restate an important result in online learning and we will use it in the following.

Lemma 1 ([58]). *For any $Q \in \mathcal{S}_+^d$ and convex feasible set $\mathcal{F} \subset \mathbb{R}^d$, suppose $u_1 = \min_{x \in \mathcal{F}} \|Q^{1/2}(x - z_1)\|$ and $u_2 = \min_{x \in \mathcal{F}} \|Q^{1/2}(x - z_2)\|$ then we have $\|Q^{1/2}(u_1 - u_2)\| \leq \|Q^{1/2}(z_1 - z_2)\|$.*

Our proof framework is similar to that of [58]. Here is a standard argument in momnet method.

Lemma 2. *Suppose $m_t = \gamma m_{t-1} + (1-\gamma)g_t$ with $m_0 = \mathbf{0}$ and $0 < \gamma < 1$. We have*

$$\sum_{t=1}^{T^*} \|m_t\|^2 \leq \sum_{t=1}^{T^*} \|g_t\|^2.$$

Proof. By Cauchy-Schwarz and Young's inequality, we have

$$\|m_t\|^2 \leq \gamma \|m_{t-1}\|^2 + (1-\gamma)\|g_t\|^2.$$

Note that $m_0 = \mathbf{0}$,

$$\frac{\|m_t\|^2}{\gamma^t} \leq (1-\gamma) \sum_{i=1}^{t} \|g_i\|^2 \gamma^{-i}.$$

So we have

$$\|m_t\|^2 \leq (1-\gamma) \sum_{i=1}^{t} \|g_i\|^2 \gamma^{t-i}.$$

Take the summation on t for both sides of the inequality, we have the conclusion. □

So we could begin to prove Theorem 2. Suppose $\{x_t\} \subset \mathbb{R}^n$. As the notation before, $x^\star = arg\min_{x\in\mathcal{F}} \sum_{t=1}^{T^*} h_t(x)$ and $x_{t+1} = \Pi_{\mathcal{F},\alpha_t^{-1/2}}(x_t - \alpha(t)\cdot m_t) = \min_{x\in\mathcal{F}} \|\alpha(t)^{-1/2}\cdot(x-(x_t-\alpha(t)\cdot m_t))\|$. By Lemma 1, $\|\alpha(t)^{-1/2}\cdot(x_{t+1}-x^\star)\|^2 \le \|\alpha(t)^{-1/2}\cdot(x_t-\alpha(t)\cdot m_t - x^\star)\|^2 = \|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2 + \|\alpha(t)^{1/2}\cdot m_t\|^2 - 2\langle q_t m_{t-1} + (1-q_t)g_t, x_t - x^\star\rangle$. Rearrange the inequality, we have

$$\begin{aligned}\langle g_t, x_t - x^\star\rangle \le \frac{1}{2(1-q_t)}\left[\|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2 - \|\alpha(t)^{-1/2}\cdot(x_{t+1}-x^\star)\|^2 + \|\alpha(t)^{1/2}\cdot m_t\|^2\right] \\ + \frac{q_t}{2(1-q_t)}\cdot\left(\|\alpha(t)^{1/2}\cdot m_{t-1}\|^2 + \|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2\right).\end{aligned} \tag{22}$$

The second inequality holds for Cauchy-Schwarz inequality and for any $a, b \in \mathbb{R}$, $ab \le \frac{a^2+b^2}{2}$.

Because $\{h_t\}_{t=1}^{T^*}$ are convex functions:

$$R_{T^*} = \sum_{t=1}^{T^*} h_t(x_t) - h_t(x^\star) \le \sum_{t=1}^{T^*} \langle g_t, x_t - x^\star\rangle$$

So we have

$$\begin{aligned}R_{T^*} \le \underbrace{\sum_{t=1}^{T^*}\left[\frac{1}{2(1-q_t)}\left[\|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2 - \|\alpha(t)^{-1/2}\cdot(x_{t+1}-x^\star)\|^2\right] + \frac{q_t}{2(1-q_t)}\|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2\right]}_{A} \\ + \underbrace{\sum_{t=1}^{T^*}\left[\frac{1}{2(1-q_t)}\|\alpha(t)^{1/2}\cdot m_t\|^2 + \frac{q_t}{2(1-q_t)}\|\alpha(t)^{1/2}\cdot m_{t-1}\|^2\right]}_{B}.\end{aligned} \tag{23}$$

First we bound the part A:

$$\begin{aligned}&\sum_{t=1}^{T^*}\left[\frac{1}{2(1-q_t)}\left[\|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2 - \|\alpha(t)^{-1/2}\cdot(x_{t+1}-x^\star)\|^2\right] + \frac{q_t}{2(1-q_t)}\|\alpha(t)^{-1/2}\cdot(x_t-x^\star)\|^2\right] \\ &\le \frac{1}{2(1-q_1)}\left[\sum_{i=1}^{n}\alpha_1^{-1}(x_1(i)-x^\star(i))^2 + \sum_{t=2}^{T^*}\sum_{i=1}^{n}(\alpha_t^{-1}-\alpha(t-1)^{-1})(x_t(i)-x^\star(i))^2 + \sum_{t=1}^{T^*}\sum_{i=1}^{n} q_t(x_t(i)-x^\star(i))^2\alpha(t)^{-1}\right]\end{aligned} \tag{24}$$

Next we bound the part B. By definition of $\alpha(t)$, $\alpha_1(1)/\sqrt{t} \le \alpha(t) \le \alpha_2(1)/\sqrt{t}$. So we have

$$\begin{aligned}&\sum_{t=1}^{T^*}\left[\frac{1}{2(1-q_t)}\|\alpha(t)^{1/2}\cdot m_t\|^2 + \frac{q_t}{2(1-q_t)}\|\alpha(t)^{1/2}\cdot m_{t-1}\|^2\right] \\ &\le \frac{\alpha_2(1)}{2(1-q_1)}\left[\sum_{t=1}^{T^*}\frac{\|m_t\|^2}{\sqrt{t}} + \sum_{t=1}^{T^*}\frac{\|m_{t-1}\|^2}{\sqrt{t}}\right] \\ &\le \frac{\alpha_2(1)}{2(1-q_1)}\left[\frac{1}{T^*}\left[\sum_{t=1}^{T^*}\|m_t\|t^{-1/4}\right]^2 + \frac{1}{T^*}\left[\sum_{t=1}^{T^*}\|m_{t-1}\|t^{-1/4}\right]^2\right] \\ &\le \frac{\alpha_2(1)}{2(1-q_1)}\left[\frac{1}{T^*}\sum_{t=1}^{T^*}\|m_t\|^2\cdot\sum_{t=1}^{T^*}t^{-1/2} + \frac{1}{T^*}\sum_{t=1}^{T^*}\|m_{t-1}\|^2\cdot\sum_{t=1}^{T^*}t^{-1/2}\right] \\ &\le \frac{\alpha_2(1)B_2^2}{(1-q_1)}\sum_{t=1}^{T^*}t^{-1/2} \\ &\le (2\sqrt{T^*}-1)\frac{\alpha_2(1)B_2^2}{(1-q_1)}.\end{aligned}$$

https://doi.org/10.52202/079017-2889

The second inequality holds for Jensen inequality and the third inequality follows from Cauchy-Schwarz inequality. The forth inequalityholds for Lemma 2.

Combine the argument above and notice that $\alpha(t)^{-1} \leq p_1^{-1} \cdot \sqrt{T^*}$, we have

$$
\begin{aligned}
R_{T^*} &\leq \frac{B_\infty^2}{2(1-q_1)} \left[n \cdot \alpha_1^{-1} + \sum_{t=2}^{T^*} n \cdot (\alpha_t^{-1} - \alpha_{t-1}^{-1}) + \sum_{t=1}^{T^*} n \cdot q_t \cdot \alpha_t^{-1} \right] + (2\sqrt{T^*} - 1) \frac{\alpha_2(1) B_2^2}{1-q_1} \\
&\leq \sqrt{T^*} \cdot \left(\frac{B_\infty^2}{2(1-q_1)} \cdot n \cdot \hat{\alpha}_{T^*}^{-1} + \frac{2 \cdot \alpha_2(1) B_2^2}{1-q_1} \right) - \frac{\alpha_2(1) B_2^2}{1-q_1} + \frac{B_\infty^2}{2(1-q_1)} \sum_{t=1}^{T^*} n \cdot q_t \cdot \alpha_t^{-1} \\
&\leq \sqrt{T^*} \cdot \left(\frac{B_\infty^2 \cdot n \cdot p_1^{-1}}{2(1-q_1)} \cdot (1 + 2q_0 q) + \frac{2 \cdot \alpha_2(1) B_2^2}{1-q_1} \right) - \frac{\alpha_2(1) B_2^2}{1-q_1}.
\end{aligned}
$$

A.4 Equivalence of ℓ_2-norm Measure and Classic Convolution in Convergence

We find that ℓ_2-norm Net is the linear transformation to the inner product convolution network, here we give the detailed calculation.

The output of the ℓ_2-norm Net in Eq. 25.

$$
Y_{\ell_2}(P_t, K) = \sqrt{\sum_{t \geq 1} \sum_{i,j} |P_t(i,j) - K(i,j)|^2} \tag{25}
$$

Therefore, we can express it as the following:

$$
\begin{aligned}
Y_{\ell_2}^2(P_t, K) &= \sum_{t \geq 1} \sum_{i,j} (P_t(i,j)^2 + K(i,j)^2 - 2P_t(i,j)K(i,j)) \\
&= \sum_{t \geq 1} \sum_{i,j} (P_t(i,j)^2 + K(i,j)^2) - \sum_{t \geq 1} \sum_{i,j} P_t(i,j)K(i,j) \\
&= \sum_{t \geq 1} \sum_{i,j} (P_t(i,j)^2 + K(i,j)^2) - 2Y_{CNN}(P_t, K).
\end{aligned} \tag{26}
$$

Notably, the term $\sum_{i,j} K(i,j)^2$ remains constant for each channel, and $\sum_{t \geq 1} \sum_{i,j} P_t(i,j)^2$ represents the square of ℓ_2-norm of each input patch. If this term is invariant across patches, the ℓ_2-norm Net's output can be regarded as a linear transformation of the CNNs' output.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: see the *abstract* and *introduction* part.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: see Sec. 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

https://doi.org/10.52202/079017-2889

Justification: see Sec.A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Instructions about experimental settings are in Sec. 5, and the URL of the project will be released after the paper is accepted.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: see the zip files of codes in the supplemental material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: see the main manuscript.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: the effect of random seed could almost be negligible since we set the same initiation seed during experiments. Reproducibility can be guaranteed.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

https://doi.org/10.52202/079017-2889

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: see the main manuscript.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: we have conformed with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: see Sec. 6.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This work poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: see the main manuscript.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

https://doi.org/10.52202/079017-2889

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This work does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.