Surface-VQMAE: Vector-quantized Masked Auto-encoders on Molecular Surfaces

Fang WU [1] **Stan Z. Li**† [1]

Abstract

Molecular surfaces imply fingerprints of interaction patterns between proteins. However, non-equivalent efforts have been paid to incorporating the abundant protein surface information for analyzing proteins' biological functions in juxtaposition to amino acid sequences and 3D structures. We propose a novel surface-based unsupervised learning algorithm termed Surface-VQMAE to overcome this obstacle. In light of surface point clouds' sparsity and disorder properties, we first partition them into patches and obtain the sequential arrangement via the Morton curve. Successively, a Transformer-based architecture named SurfFormer was introduced to integrate the surface geometry and capture patch-level relations. At last, we enhance the prevalent masked auto-encoder (MAE) with the vector quantization (VQ) technique, which establishes a surface pattern codebook to enforce a discrete posterior distribution of latent variables and achieve more condensed semantics. Our work is the foremost to implement pretraining purely on molecular surfaces and extensive experiments on diverse real-life scenarios including binding site scoring, binding affinity prediction, and mutant effect estimation demonstrate its effectiveness. The code is available at `https://github.com/smiles724/VQMAE`.

1. Introduction

Proteins are critical components of biological systems, performing a diverse range of functions that support a broad spectrum of cellular processes and biological pathways. These complex macromolecules exhibit intricate, multifaceted characteristics, which can be expressed through various paradigms such as amino acid sequences, 3D structures, and the distinctive attributes of surface regions. The past decade has witnessed progress in high-throughput sequencing (HTS) technologies, cryogenic electron microscopy, and sophisticated protein structure prediction algorithms (Jumper et al., 2021), which resulted in an explosion of publicly accessible protein sequences and structures. To leverage the large volume of those databases, many self-supervised learning (SSL) studies have been introduced via mechanisms like contrastive learning (Hermosilla & Ropinski, 2022; Zhang et al., 2022b), denoising diffusion (Liu et al., 2023; Zhang et al., 2023), and self-prediction (Zhang et al., 2022b; Wu et al., 2022a; 2023d; Lee et al., 2023; Wu et al., 2023b;a).

Yet, many vital biological processes, including membrane transport and cell signaling, are governed by intricate networks of protein-protein interactions (PPIs) rather than single proteins acting independently (Lin et al., 2024). In contrast to sequence and structure, a protein's molecular surface, which is characterized by a compact and smooth composition of atoms at its boundary and showcases both chemical and geometric features, holds a more direct relevance to biomolecular interactions and function (Mylonas et al., 2021; Riahi et al., 2023). Despite that, current efforts in protein representation learning predominantly concentrate on encoding amino acid sequences (Rives et al., 2021; Lin et al., 2022) and 3D structural elements (Zhang et al., 2022b), overlooking the indispensable part of surfaces.

Conducting self-supervision on molecular surfaces is not easy due to numerous reasons. Firstly, the information density of surface point clouds is sparse compared to languages or images. Instead, they are natural signals with heavy redundancy thereby the pretraining tasks can be accomplished even without holistic understanding (Chen et al., 2023). Secondly and intuitively, surface point clouds are inherently disordered and do not have a sequential arrangement like words in a sentence or pixels in a picture. Moreover, it is challenging to bridge the gap between the pretraining and fine-tuning stages. Even though the popular masking and reconstruction technique (Lin et al., 2022; Pang et al., 2022; Zhang et al., 2022b; Luo et al., 2023) exhibits deep comprehension, no restriction has been imposed in the latent space between the encoder and the decoder. On the contrary,

[1]School of Engineering, Westlake University. Correspondence to: Stan Z. Li <stan.zq.li@westlake.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

existing algorithms usually resort to a universal learnable embedding to represent the masked tokens, which may capture latent representations with a lower semantic level than downstream problems.

In response to this gap, we introduce a new family of unsupervised methods named Surface-VQMAE with several key innovations for apprehending more effective surface representations. Firstly, concerning its inherent sparsity and disorder properties, our algorithm partitions the input surface point cloud into multiple irregular patches and organizes them by the Morton curve. Secondly, a SurfFormer architecture is proposed to grasp the interactions among distinct patches and gain a global point cloud understanding. It extends the vanilla attention mechanism (Vaswani et al., 2017) by incorporating geometric structural information. Last but not least, we successfully combine the prevailing masked auto-encoder (MAE) framework (He et al., 2022) with discrete latent representations by enforcing and parameterizing the posterior distribution of latent variables to be categorical, which facilitates the feature extractor to acquire more condensed semantics. We examine the generalization and robustness of Surface-VQMAE on diverse protein engineering tasks. In the binding site evaluation task, Surface-VQMAE achieves AUROC and a balanced accuracy of 94.78% and 87.16%, respectively. In the antibody-antigen affinity prediction task, it outweighs all potential-based and deep learning-based (DL) approaches. In the mutant effect estimation task, Surface-VQMAE also attains new state-of-the-art (SOTA) performance on nearly all metrics. All evidence indicates the promise of surface-based DL, and we envision more efforts to investigate its potential in a wider range of real-world applications.

2. Related Works

2.1. Protein Surface Representation

Over the past few years, a cluster of DL algorithms, inspired by advancements in the fields of language and vision, has emerged to harness the wealth embedded in protein sequences and structures (Wu et al., 2022a; Lin et al., 2022; Zhang et al., 2022b; Wu et al., 2022b). Simultaneously, growing interests have been attracted to integrate surface-related data to enhance protein representation learning (Leem et al., 2022; Lee et al., 2023). From a unique viewpoint, the characteristics of the molecular surface dictate the type and the strength of the interactions that a protein can have with other molecules. It is defined based on van der Waals (vdW) radii(Connolly, 1983) and is commonly represented as meshes derived from signed distance functions. MaSIF (Leem et al., 2022) pioneered the use of mesh-based geometric DL to abstract the internal parts of the protein fold and delved into the analysis of protein interactions. A subsequent study (Sverrisson et al., 2021) removed its high pre-computation costs of featurization and demonstrated competitiveness by modeling molecular surfaces as point clouds with atom categories assigned to each point. Other seminal works attempted to connect protein surfaces with structures in a multi-modality manner (Somnath et al., 2021) and conduct comprehensive pretraining strategies (Lee et al., 2023). Nevertheless, none of the aforementioned studies have rigorously explored SSL immediately on the informative molecular surfaces.

2.2. 3D Point Cloud Pretraining

Geometric DL is an umbrella term encompassing emerging techniques that generalize neural networks to Euclidean and non-Euclidean domains like manifolds and meshes. Point clouds, a native representation of 3D data from range sensors, have recently gained popularity in representing surfaces, and relevant SSL techniques have also been widely studied. Among them, the contrastive kind has been extensively investigated including DepthContrast (Zhang et al., 2021) and CrossPoint (Afham et al., 2022). Another line of research demonstrates the superiority of masked point modeling (MPM) mechanisms. For instance, PointMAE (Pang et al., 2022) extended the masked auto-encoder (MAE) by randomly masking point patches and recovering masked regions. PointM2AE (Zhang et al., 2022a) additionally utilized a hierarchical Transformer and designed a corresponding masking strategy. PointGPT (Chen et al., 2023) exploited an auto-regressive fashion to reconstruct the masked patches. However, insufficient attention has been given to the imposition of constraints on the representation space of latent variables during pretraining, let alone assessing their effectiveness in the context of molecular surfaces.

3. Method

3.1. Preliminaries of Protein Surface

Surface Generation. There are currently two distinct approaches to generating protein surfaces. The first, exemplified by MaSIF (Gainza et al., 2020), relies on the MSMS program (Sanner et al., 1996) to obtain meshes and handcrafted features (Mylonas et al., 2021; Stebliankin et al., 2023; Li & Liu, 2023). However, it is time-consuming and demands substantial memory resources. In contrast, the second kind (Sverrisson et al., 2021) circumvents this obstacle by adopting a fast sampling-based mechanism to attain the surface point cloud, allowing for an end-to-end training fashion.

It is assumed that a protein comprises either a set of atoms $\mathcal{V}^a = \{v_i^a\}_{i=1}^N$ at the fine-grained level or residues $\mathcal{V}^R = \{v_i^R\}_{i=1}^{N'}$ at the coarse-grained level. Each atom v_i^a has a corresponding chemical type $\mathbf{t}_i^a \in \mathbb{R}^6$ in the list $[\mathrm{C}, \mathrm{H}, \mathrm{O}, \mathrm{N}, \mathrm{S}, \mathrm{Se}]$ encoded as one-hot vectors and 3D co-

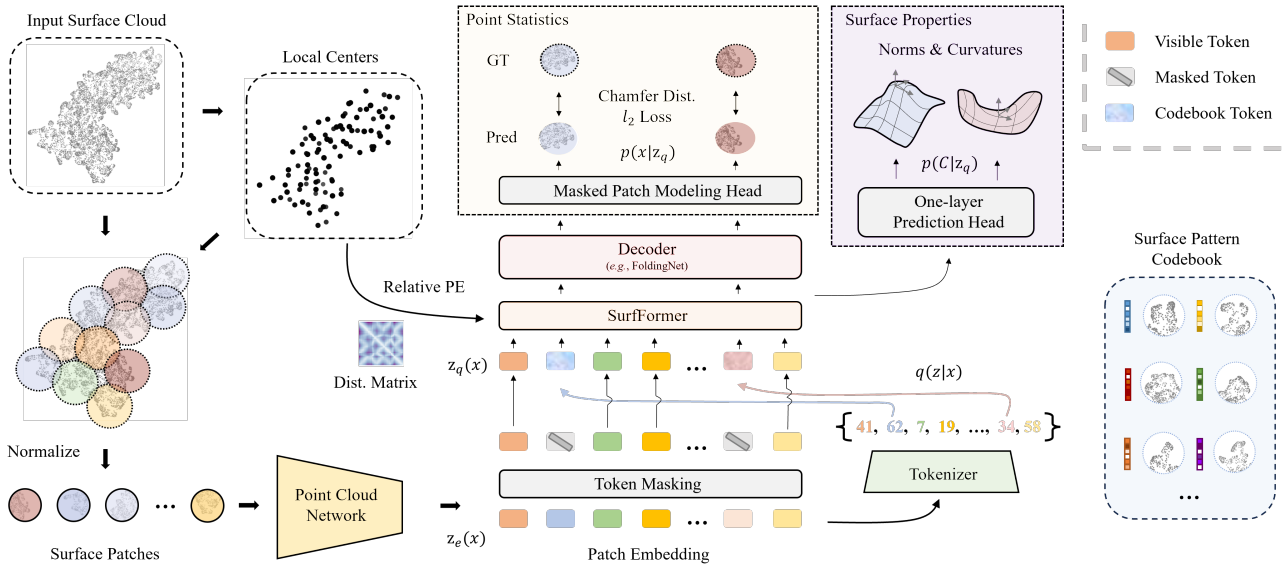

Figure 1. The overall pipeline of our proposed Surface-VQMAE. The input surface point cloud is first preprocessed into patches through the farthest point search (FPS) algorithm and acquires the sequential order via Morton code. It is further fed into a point cloud network to aggregate patch representations. Then we randomly mask a portion of patches and replace them with relaxed codebook vectors. Both visible token embeddings and sampled codebook vectors are forwarded to SurfFormer to gain a global point cloud understanding. Finally, two pre-text tasks are proposed to reconstruct the center point coordinates and forecast surface curvatures.

ordinates $\mathbf{x}_i^a \in \mathbb{R}^3$. Then we describe the protein surface as the level set of a smooth distance function to the atom centers. To be specific, we first randomly draw an upsampling ratio of $\eta = 20$ points from a normal distribution $\mathcal{N}(\mu = \mathbf{x}_i^a, \sigma = 10\text{Å})$ for each atom center, obtaining $\{\mathbf{x}_i^s\}_{i=1}^{\eta N}$. Then we associate each atom type $\mathbf{t}_i^a$ with a different radius σ_i^a and define the smooth distance function with a stable log-sum-exp reduction as:

$$\text{SDF}(\mathbf{x}_i^s) = -f(\mathbf{x}_i^s) \cdot \log \sum_{j=1}^{N} \exp(-\|\mathbf{x}_i^s - \mathbf{x}_j^a\|/\sigma_j^a), \quad (1)$$

$$f(\mathbf{x}_i^s) = \frac{\sum_{j=1}^{N} \exp(-\|\mathbf{x}_i^s - \mathbf{x}_j^a\|)\sigma_j^a}{\sum_{j=1}^{N} \exp(-\|\mathbf{x}_i^s - \mathbf{x}_j^a\|)}, \quad (2)$$

where $f(.)$ serves as the average atom radius in a neighborhood of any surface point $\mathbf{x}_i^s \in \mathbb{R}^3$. After that, we let these points converge towards this target level set (*i.e.*, Equ. 1) by gradient descent. That is, we derive the level set surface at a radius of $r = 1.05$Å by minimizing the following squared loss function:

$$E\left(\mathbf{x}_1^s, \ldots, \mathbf{x}_{\eta N}^s\right) = \frac{1}{2} \sum_{i=1}^{\eta N} \left(\text{SDF}\left(\mathbf{x}_i^s\right) - r\right)^2. \quad (3)$$

Subsequently, we remove points trapped inside the protein and leverage the normalized gradient of the distance function SDF(.) at location $\mathbf{x}_i^s$ as the normal vector $\mathbf{n}_i^s$. Finally, we denote the pseudo protein surface as an oriented point cloud $S = \{\mathbf{x}_i^s, \mathbf{n}_i^s\}_{i=1}^{M}$. Here we resort to the KeOps library (Charlier et al., 2021) to implement this sampling strategy efficiently on protein batches.

Last but not least, we employ a simple geometric aggregation network (GeoAN) to attain the chemical features of each point $\mathbf{h}_i^s \in \mathbb{R}^{\phi_h}$. Particularly, for each surface point $\mathbf{x}_i^s$, we find the $\zeta = 16$ nearest residues centers (C_α positions) and their related residue types $\{\mathbf{x}_i^R, \mathbf{t}_i^R\}_{i=1}^{\zeta}$. Then we use an embedding layer to map $\mathbf{t}_i^R$ to its embedding $\mathbf{e}_i^R \in \mathbb{R}^{\phi_R}$ and apply a multi-layer perception (MLP) to the vectors $\mathbf{e}_i^R \oplus 1/||\mathbf{x}_i^s - \mathbf{x}_i^R||^2$. After that, we perform an average pooling over these transformed vectors (*i.e.*, $i = 1, ..., \zeta$) and append a second MLP to compute $\mathbf{h}_i^s$. Notably, we collect the residue-level instead of atom-level internal information (Sverrisson et al., 2021) for the sake of $\mathbf{h}_i^s$ and observe slightly better empirical results.

Surface Patch Partition. A naive approach treats per point as one token. However, such point-wise reconstruction pretrain task tends to have unbearable computational cost due to the quadratic complexity of self-attention (Yu et al., 2022). Considering the inherent sparsity of surface point clouds, we utilize the farthest point sampling (FPS) and the K-nearest neighbors (KNN) algorithm to acquire center points and point patches. To be explicit, we initially sample center points $\mathbf{X}^c$ via FPS with a downsampling ratio of ρ.

Then ρM point patches $\mathbf{X}^{\mathrm{p}}$ are constructed by selecting the K nearest points from $\mathbf{X}^s$ for each center point. The partitioning procedure is formulated as:

$$\mathbf{X}^{\mathrm{c}} = \mathrm{FPS}(\mathbf{X}^s), \quad \mathbf{X}^{\mathrm{c}} \in \mathbb{R}^{\rho M \times 3}, \tag{4}$$

$$\mathbf{X}^{\mathrm{p}} = \mathrm{KNN}\left(\mathbf{X}^{\mathrm{c}}, \mathbf{X}^s\right), \quad \mathbf{X}^{\mathrm{p}} \in \mathbb{R}^{\rho M \times K \times 3}. \tag{5}$$

Notably, voxelization (Choy et al., 2019; Tian et al., 2023) is another common option to divide point clouds into patches. It projects the point clouds to 3D voxel grids and encodes features of points inside the same voxel. However, it ignores the rotation-translation equivariance and point densities can change significantly in different voxels.

Sorting. Unlike 1D protein sequences, a point cloud lacks sequential arrangement. To address this inherent disorder dilemma, we borrow ideas from PointGPT (Chen et al., 2023) and organize the point patches based on a geometric ordering, namely the Morton-order curve (Morton, 1966). Concretely, the coordinates of the center points $\mathbf{X}^{\mathrm{c}}$ are first encoded into one-dimensional space via Morton code, where the order is determined. Then the point patches are arranged in the same order and the procedure is written as follows:

$$\mathcal{O} = \mathrm{argmax}\left(\mathrm{Morton\,Code}\left(\mathbf{X}^{\mathrm{c}}\right)\right), \quad \mathcal{O} \in \mathbb{R}^{\rho M \times 1}, \tag{6}$$

This order $\mathcal{O}$ not only introduces sequential properties to point patches $\mathbf{X}^{\mathrm{p}}$ but also preserves the essential local structures (Chen et al., 2023).

3.2. Backbone Architecture

To align with the hierarchical granularity of protein surfaces (*i.e.*, point-level, and patch-level), we propose a hierarchical scheme to capture varying degrees of surface detail. Towards this goal, a point cloud network (*e.g.*, quasi-geodesic convolution networks) is first utilized to extract fine-grained and local (*i.e.*, point-wise) surface representations. Then SurfFormer is followed to excavate coarse-grained and global (*i.e.*, patch-wise) surface representations.

Point Cloud Network. We begin with the estimation of a local coordinate system $(\mathbf{n}_i^s, \mathbf{u}_i^s, \mathbf{o}_i^s)$. Notably, the surface point normal $\mathbf{n}_i^s$ has already been gained using the gradient of Equ. 1, and we smooth this vector field by a Gaussian kernel with $\sigma_{\mathbf{n}}$. Mathematically, $\mathbf{n}_i^s = \mathrm{Normalize}\left(\sum_{j=1}^{M} \exp\left(-\left\|\mathbf{x}_i^s - \mathbf{x}_j^s\right\|^2 / 2\sigma_{\mathbf{n}}^2\right) \mathbf{n}_j^s\right)$.

Then we compute the tangent vectors $\mathbf{u}_i^s$ and $\mathbf{o}_i^s$ using the efficient formula in Duff et al. (2017). To be explicit, let $\mathbf{n}_i^s = [\chi_1, \chi_2, \chi_3]$ be a unit vector, $a = -1/(c + \chi_3)$, $b = a\chi_1\chi_2$, and $c = \mathrm{sign}(\chi_3)$, then we get:

$$\mathbf{u}_i^s = \left[1 + ac\chi_1^2, bc, -c\chi_1\right], \quad \mathbf{o}_i^s = \left[b, a\chi_2^2 + c, -\chi_2\right]. \tag{7}$$

In a later stage, we approximate the geodesic distance between two surface points $\mathbf{x}_i^s$ and $\mathbf{x}_j^s$ with their unit normals as:

$$\mathrm{d}_{ij} = \left\|\mathbf{x}_i^s - \mathbf{x}_j^s\right\| \cdot \left(2 - \langle \mathbf{n}_i, \mathbf{n}_j \rangle\right). \tag{8}$$

The filters are then localized by a smooth Gaussian window of radius σ_{d}, leading to $w\left(\mathrm{d}_{ij}\right) = \exp\left(-\mathrm{d}_{ij}^2 / 2\sigma_{\mathrm{d}}^2\right)$. In the neighborhood of any surface point $\mathbf{x}_i^s$, we define a vector that encodes the relative position and orientation of neighbor points $\mathbf{x}_j^s$ in the local coordinate system $(\mathbf{n}_i^s, \mathbf{u}_i^s, \mathbf{o}_i^s)$ as:

$$\mathbf{p}_{ij} = \left(\mathbf{x}_i^s - \mathbf{x}_j^s\right)^\top \cdot \left[\mathbf{n}_i^s \oplus \mathbf{u}_i^s \oplus \mathbf{o}_i^s\right]. \tag{9}$$

Finally, the quasi-geodesic convolution is operated using a trainable MLP to weigh features in a geodesic neighborhood of the local reference surface point $\mathbf{x}_i^s$. The formula of the l-th layer to update the vector signal is therefore:

$$\mathbf{h}_i^{s(l+1)} = \sum_{j \in \mathcal{N}(i)} w(\mathrm{d}_{ij}) \mathrm{MLP}(\mathbf{p}_{ij}) \mathbf{h}_j^{s(l)}, \tag{10}$$

where $\mathcal{N}(i)$ is the i's geodesic neighborhood set determined by the filter window size σ_{d}. We stack L_1 layers of this geodesic convolution as the point cloud network and note that the choice of intermediate layers is completely free. We envision future efforts to replace geodesic convolutions (Sverrisson et al., 2021) with some other categories of advanced 3D point cloud algorithms such as PointNet++ (Qi et al., 2017) and PointMLP (Ma et al., 2022).

SurfFormer. At the last layer of the point cloud network, we perform a max-pooling operation and a MLP to aggregate initial patch features as $\mathbf{h}_i^{\mathrm{p}} = \mathrm{MLP}\left(\mathrm{Pool}\left(\left\{\mathbf{h}_j^{s(L_1)} \middle| \mathbf{x}_j^s \in \mathbf{X}_i^{\mathrm{p}}\right\}\right)\right) \in \mathbb{R}^{\phi_{\mathrm{p}}}$, accompanied by a modified Transformer (Vaswani et al., 2017) dubbed SurfFormer to capture pairwise interactions between patches. SurfFormer has L_2 layers and each layer consists of standard layer-norm, feed-forward blocks, and a specially designed geometric multi-head self-attention to learn global correlations in feature and geometric spaces among surface patches. Particularly, the output patch feature at the l-th layer is the weighted sum of all projected input ones as $\mathbf{h}_i^{\mathrm{p}(l+1)} = \sum_{j=1}^{\rho M} \alpha_{ij} \cdot \mathbf{h}_j^{\mathrm{p}(l)} \mathbf{W}_V$. Here, the weight coefficient α_{ij} is calculated by a row-wise softmax on the attention score as:

$$\alpha_{ij} = \mathrm{Soft.}\left(\frac{\left(\mathbf{h}_i^{\mathrm{p}(l)} \mathbf{W}_Q\right)\left(\mathbf{h}_i^{\mathrm{p}(l)} \mathbf{W}_K + \mathbf{r}_{ij} \mathbf{W}_G\right)^\top}{\sqrt{\phi_{\mathrm{p}}}}\right), \tag{11}$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V, \mathbf{W}_G \in \mathbb{R}^{\phi_{\mathrm{p}} \times \phi_{\mathrm{p}}}$ are all projection matrices. Besides, $\mathbf{r}_{ij} \in \mathbb{R}^{\phi_{\mathrm{p}}}$ is a geometric structural embedding. It is obtained by expanding the geodesic distance d_{ij} with radial basis functions (Schütt et al., 2018) as:

$$\mathbf{r}_{ij,k} = \exp\left(-\gamma\left(\|\mathrm{d}_{ij} - \mu_k\|^2\right)\right), \tag{12}$$

which is located at different centers $0\text{Å} \leq \mu_k \leq 30\text{Å}$ every $\frac{30}{\phi_{\mathrm{p}}}$Å with $\gamma = 10$Å. $\mathbf{r}_{ij}$ can be perceived as an invariant of relative positional encoding (RPE) (Shaw et al., 2018) and its design satisfies the 3D roto-translational invariance. Moreover, the order $\mathcal{O}$ gained from Equ. 6 enables the assignment of sinusoidal positional embedding (Vaswani et al., 2017) to $\mathbf{X}^{\mathrm{p}}$ and we add this absolute positional encoding (APE) to every SurfFormer block to enhance a global point cloud understanding (Chen et al., 2023).

3.3. VQ-MAE

Relying less on human-annotated data, SSL has significantly advanced various domains including language, vision, and AI for life science (Wu et al., 2022a). Among them, masked auto-encoder (MAE) (He et al., 2022; Pang et al., 2022) is a promising scheme that randomly masks a portion of input data and takes advantage of an auto-encoder to reconstruct explicit or implicit features corresponding to the original masked content (Li et al., 2023).

Masking. Considering surface patches may overlap, we mask them separately so that information is kept complete in each patch. Besides, we constrain $\rho K = 1$ to make a trade-off between covering the entire point cloud and minimizing the patch overlap. With a masking ratio δ, the coordinate sets of masked and visible patches are denoted as $\mathbf{X}^{\mathrm{p,m}} \in \mathbb{R}^{\delta\rho M \times K \times 3}$ and $\mathbf{X}^{\mathrm{p,vis}} \in \mathbb{R}^{(1-\delta)\rho M \times K \times 3}$, respectively. Empirical practice finds that a masking strategy at a high ratio ($\delta \geq 50\%$) works perfectly in spite of the uneven density in surface point clouds.

Tokenization. For each masked patch token in $\mathbf{X}^{\mathrm{p,m}}$, rather than using a shared learnable mask embedding, we replace its representation $\mathbf{h}_i^{\mathrm{p,m}}$ with a code embedding $\mathbf{e}$ through the vector quantization (VQ) technique. Remarkably, the vanilla VQ (Van Den Oord et al., 2017) faces several intrinsic flaws. For instance, the evidence lower bound (ELB) becomes difficult to optimize as the posterior distribution $q(.)$ is a discrete distribution. Though attempts have been made to address this issue, such as an online cluster assignment procedure in conjunction with the straight-through estimator (Van Den Oord et al., 2017), the optimization remains sub-optimal. More importantly, it is better to express some uncertainty over latent vectors as the posterior distribution $q(.)$ does not generally need to be deterministic. As a consequence, we draw inspiration from dVAE (Ramesh et al., 2021) and enable retrieving vectors anywhere in the convex hull of codebook vectors.

Specifically, we define a codebook $\mathcal{Q} = \{(i, \mathbf{e}(i))\}_{i=1}^{N_B}$ with size N_B as the set of finite pairs of code i and its code embedding $\mathbf{e}(i) \in \mathbb{R}^{\phi_{\mathrm{p}}}$. Then the sampled latent vector $\mathbf{z}_i^{\mathrm{p,m}}$ is then a weighted sum of these codebook vectors as:

$$\mathbf{z}_i^{\mathrm{p,m}} = \sum_{i=1}^{N_B} \frac{\exp\left(\frac{g_i + \log\left(q\left(\mathbf{e}_j | \mathbf{h}_i^{\mathrm{p,m}}\right)\right)}{\tau}\right)}{\sum_{j=1}^{N_B} \exp\left(\frac{g_j + \log\left(q\left(\mathbf{e}_j | \mathbf{h}_i^{\mathrm{p,m}}\right)\right)}{\tau}\right)} \cdot \mathbf{e}_i. \quad (13)$$

This re-parameterization trick is realized via Gumble Softmax relaxation (Jang et al., 2016) and g_i is an identical and independent sample from the Gumble distribution. As the temperature $\tau \to 0$, the relaxation becomes tight.

Decoding and Prediction Targets. At last, we merge visible token embeddings and sampled codebook vectors from the relaxed posterior together as $\mathbf{H}^{\mathrm{p}(0)} = \mathbf{H}^{\mathrm{p,vis}} \oplus \mathbf{Z}^{\mathrm{p,m}} \in \mathbb{R}^{\rho M \times \phi_{\mathrm{p}}}$ and forward them to SurfFormer and decoders for the final pre-text tasks. This VQ masking approach successfully avoids early leakage of location information to SurfFormer, whose output is denoted as $\mathbf{H}^{\mathrm{p}(L_2)}$.

Here, we introduce two categories of prediction targets, *i.e.*, point statistics, and surface properties, to guide model learning geometric characteristics of surface clouds. The first type is conventionally reconstructing the coordinates of center points $\mathbf{X}^{\mathrm{c,m}}$ for each masked surface patch. Following prior works (Li et al., 2023; Chen et al., 2023), a simple MLP is used to project to a vector that has the same number of dimensions as the total number of coordinates in a point patch. Then followed by a reshape operation, the predicted positions for masked surface patches are acquired as:

$$\hat{\mathbf{X}} = \mathrm{Reshape}\left(\mathrm{MLP}\left(\mathbf{H}^{\mathrm{p}(L_2)}\right)\right), \hat{\mathbf{X}} \in \mathbb{R}^{\delta\rho M \times K \times 3}. \quad (14)$$

The second target is based on the 3D shape geometry, and we select surface curvatures as they can be calculated in a closed form from local points (Tian et al., 2023). For each masked surface patch $\mathbf{X}_i^{\mathrm{p,m}} \in \mathbb{R}^{K \times 3}$ with its center point $\mathbf{x}_i^{\mathrm{c,m}} \in \mathbb{R}^3$, we first compute a covariance matrix:

$$\mathbf{\Sigma} = \frac{1}{k} \sum_{\mathbf{x}_{i,j}^{\mathrm{p,m}} \in \mathbf{X}_i^{\mathrm{p,m}}} \mathbf{x}_{i,j}^{\mathrm{p,m}} \mathbf{x}_{i,j}^{\mathrm{p,m}\top} - \mathbf{x}_i^{\mathrm{c,m}} \mathbf{x}_i^{\mathrm{c,m}\top}, \quad \mathbf{\Sigma} \in \mathbb{R}^{3 \times 3}. \quad (15)$$

Then after the eigen-decomposition of $\mathbf{\Sigma}$ (*e.g.*, singular value decomposition or eigenvalue decomposition), eigenvalues can be attained as ϵ_1, ϵ_2, and ϵ_3. The three pseudo curvatures vectors $\boldsymbol{\psi} = \{\psi_i\}_{i=1}^3$ can be therefore computed as (Mitra & Nguyen, 2003):

$$\psi_i = \frac{\epsilon_i}{\sum_{j=1}^3 \epsilon_j}, \quad i \in \{1, 2, 3\}. \quad (16)$$

Similarly, another MLP is appended to $\mathbf{H}^{\mathrm{p}(L_2)}$ to forecast curvatures, denoted as $\hat{\boldsymbol{\psi}}$. It can be proved that this curvature target is roto-translation invariant (see Appendix A). Noticeably, we merely employ two lightweight decoders (*i.e.*,

MLPs) for different targets separately, to enforce the encoder to embed more semantics of the surface point clouds (Zhang et al., 2022a). We tried heavyweight decoders such as FoldingNet (Yang et al., 2018) to reconstruct the sub-clouds but observed no discernible improvements.

Training Losses. The overall training objective is constituted of three parts: the typical losses to recover the coordinates and curvatures for each surface patch and the Kullback-Leibler (KL) divergence to approximate the desired latent distribution $p(.)$. Rigorously, the total loss $\mathcal{L}$ can be written as:

$$\begin{aligned}\mathcal{L} =& \nu_1 \mathcal{L}_{\text{rec}}\left(\mathbf{X}^{\text{p,m}}, \hat{\mathbf{X}}\right) + \nu_2 \mathcal{L}_{\text{cur}}\left(\boldsymbol{\psi}, \hat{\boldsymbol{\psi}}\right) \\ &+ \nu_3 \mathcal{L}_{\text{KL}}\left(q\left(\mathbf{Z}^{\text{p,m}} | \mathbf{H}^{\text{p,m}}\right), p(\mathbf{Z}^{\text{p,m}})\right)\end{aligned} \tag{17}$$

where ν_1, ν_2, and ν_3 are pre-defined hyperparameters to balance the weights of different loss terms. $p(.)$ is the prior on the latent space and is usually initialized to a uniform distribution over all codebook vectors. $\mathcal{L}_{\text{cur}}(.)$ is supervised via a root mean squared error (RMSE). Meanwhile, the reconstruction loss $\mathcal{L}_{\text{rec}}(.)$ is formulated using the l_2-norm Chamfer distance (Fan et al., 2017) as:

$$\begin{aligned}\mathcal{L}_{\text{rec}} = \frac{1}{\delta\rho M K} \sum_{i=1}^{\delta\rho M} \Bigg(& \sum_{a \in \hat{\mathbf{X}}_i} \min_{b \in \mathbf{X}_i^{\text{p,m}}} \|a-b\|_2^2 \\ &+ \sum_{a \in b \in \mathbf{X}_i^{\text{p,m}}} \min_{\hat{\mathbf{X}}_i} \|a-b\|_2^2 \Bigg).\end{aligned} \tag{18}$$

4. Experiments

To verify the effectiveness of Surface-VQMAE, we examine three crucial real-world PPI applications. More experimental details are explained in the Appendix B.

Pretraining Data. The unlabeled data for pretraining Surface-VQMAE is procured from PDB-REDO (Joosten et al., 2014). It contains refined X-ray structures in PDB, and we cluster protein chains based on 50% sequence identity as Luo et al. (2023), which leads to 38,413 chain clusters. These clusters are further randomly divided into the training, validation, and test sets by 95%/0.5%/4.5%, respectively.

4.1. Protein Interface Scoring

Problem Statement and Background. The arrival of successful monomer folding algorithms such as AlphaFold (Jumper et al., 2021) marks a significant advancement in multimer folding. Leveraging them alongside molecular docking tools allows for predicting complexes when the structures of individual target proteins are known, a crucial step in drug, vaccine, and therapeutic development. However, modern docking computational tools (Andrusier et al., 2007; Leman et al., 2020) often yield numerous candidate complexes with seemingly favorable binding scores that ultimately fail confirmation in laboratory settings. This discrepancy underscores the pressing need for robust scoring functions. DL emerges as a promising avenue to assess and rank the binding strength of the predicted protein complex.

Dataset and Metrics. Following Stebliankin et al. (2023), PPI pairs are taken from the PRISM list of nonredundant proteins (Baspinar et al., 2014), the ZDock benchmark (Vreven et al., 2015), PDBBind (Liu et al., 2015), and Structural Antibody Database (SAbDab) (Dunbar et al., 2014). The train and test splits are based on sequence and structural similarity. Concretely, sequence splits are performed using CD-HIT60, and structural splits are performed using TM-align. This results in 2,958 and 356 proteins, respectively, and 10% of the training set is reserved for validation. We pick up six metrics for comparison, containing area under the receiver operating characteristic (AUROC), average precision (AP), balanced accuracy (BAcc.), F1 score, precision, and recall.

Baselines. Two sorts of benchmarks are selected for fair comparison. The first category contains seven empirical-based tools. Among them, **FireDock** (Andrusier et al., 2007), **PyDock** (Cheng et al., 2007), **RosettaDock** (Leman et al., 2020), and **ZRANK2** (Pierce & Weng, 2008) are energy-based methods, whose binding score is defined as the weighted sum of energy terms. Meanwhile, **AP-PISA** (Viswanath et al., 2013), **SIPPER** (Pons et al., 2011) and **CP-PIE** (Ravikant & Elber, 2010) are potential-based algorithms, which involve computing the atomic and residue-level iteration properties including frequency of interaction types and solvent-accessible surface area (SASA). In addition, three ML-based mechanisms are considered. **MaSIF** (Leem et al., 2022) pioneers the use of mesh-based geometric DL to predict protein interactions. **dMaSIF** (Sverrisson et al., 2021) extends MaSIF by bypassing the precomputation of physico-chemical features and instead calculating molecular surfaces directly from the atomic point cloud in real time. **PIsToN** (Stebliankin et al., 2023) adapts a vision Transformer (ViT) for protein binding assessment.

Results. Table 1 documents the outcomes, where the best and second-best performance is in bold and underlined respectively. The superiority of Surface-VQMAE is evident from its outstanding AUROC score of 0.9583, which is 1.03% than the next best method PIsTON. Additionally, Surface-VQMAE outperforms all other models in terms of AP (0.9503), BAcc. (87.16%), and F1-score (0.8709). This phenomenon illustrates that Surface-VQMAE can rank the native binding complexes more accurately than existing energy-based, potential-based, and DL tools.

Table 1. Classification performance of protein binding scoring functions, where red cells stand for surface-based algorithms. Scores for the baselines are directly reported from PIsToN (Stebliankin et al., 2023).

Method	AUROC	AP	BAcc	F1	Precision	Recall
SIPPER	74.52	72.23	68.64	70.32	66.80	74.23
RosettaDock	75.28	82.77	76.06	69.81	**94.50**	55.34
ZRANK2	83.56	87.37	77.89	74.36	88.48	64.13
PyDock	86.11	87.17	77.66	77.62	77.79	77.45
CP-PIE	88.11	89.28	81.03	79.94	84.87	75.55
AP-PISA	89.85	90.30	81.98	81.28	84.63	78.18
FireDock	89.95	90.54	81.83	81.58	82.81	80.38
MaSIF	82.52	83.96	74.93	72.51	80.25	66.13
dMaSIF	89.81	89.67	81.34	81.60	80.49	82.74
PIsToN	93.55	94.41	85.25	84.59	88.55	80.97
Surface-VQMAE	**94.78**	**95.03**	**87.16**	**87.09**	90.92	**83.58**

4.2. Antibody-antigen Binding Prediction

Problem Statement and Background. Accurate prediction of protein–ligand binding affinity (ΔG) has a variety of applications including antibody design in immunotherapy, enzyme engineering for reaction optimization, and construction of biosensors (Guo & Yamaguchi, 2022). It can greatly lower the overall cost of drug discovery in structure-based drug design (Wu et al., 2023c).

Dataset and Metrics. The test data come from SAbDab (Dunbar et al., 2014) with 4,883 antibody-antigen complexes after removing duplicates and structures without antigens. Among them, 566 instances have binding affinity labels and are used as the test set. Our training set comes from Myung et al. (2022) with 197 complexes after removing instances appeared in the test set. Then they are randomly split into two halves with a validation ratio of 50%. Following Jin et al. (2023), we emulate a more realistic scenario and predict the structure of all antibody-antigen complexes in the test set using ZDock (Vreven et al., 2015). Here, only the Pearson correlation is employed as the evaluation metric.

Baselines. We compare our method with two groups. The first is physic-based potentials, including ZRANK (Pierce & Weng, 2007), ZRANK2 (Pierce & Weng, 2008), RosettaDock (Leman et al., 2020), PyDock (Cheng et al., 2007), SIPPER (Pons et al., 2011), AP-PISA (Viswanath et al., 2013), CP-PIE (Ravikant & Elber, 2010), FireDock, and FireDock AB (Andrusier et al., 2007). The second is a ML-based approach, *i.e.*, Neural Euler's Rotation Equations (NERE) (Jin et al., 2023), which predicts a rotation by modeling the force and torque between protein and ligand atoms.

Results. We inspect the capability of Surface-VQMAE in two circumstances, namely, crystal and docked structures. As shown in Table 2, our model outperforms existing physics-based potentials and NERE significantly with an average increase of 36.9%. It can also be observed that the Pearson correlations decrease tremendously for all models when transferring from ground truth to predicted structures. However, Surface-VQMAE is less sensitive and more robust to this docking error than NERE.

Table 2. Performance of binding affinity estimation (ΔG) on the SAbDab test set, where yellow and red cells indicate structure-based and surface-based algorithms separately. Scores for the baselines are directly reported from NERE (Jin et al., 2023).

	Crystal	ZDock
SIPPER	-0.138	0.003
RosettaDock	0.064	0.025
ZRANK	0.318	0.163
ZRANK2	0.176	0.151
PyDocK	0.248	0.164
CP-PIE	0.234	0.120
AP-PISA	0.323	0.144
FireDock	0.101	-0.052
FireDock-AB	0.199	0.042
NERE	0.340	0.234
Surface-VQMAE	**0.411**	**0.358**

4.3. Mutant Effect Prediction

Problem Statement and Background. Antibodies play a crucial role in recognizing and binding to proteins present on the pathogen surfaces, initiating immune responses through interactions with receptor proteins in immune cells. So, it is essential to develop methods to regulate these interactions. A common strategy for manipulation is to mutate amino acids at the interface, and computational techniques are necessary to guide the identification of beneficial mutations by predicting their effects on binding strength. This is often assessed through the change in binding free energy, (*i.e.*, $\Delta\Delta G$).

Table 3. Evaluation of mutant effect prediction ($\Delta\Delta G$) on the SKEMPI.v2 dataset, where yellow, blue, and red cells are sequence-based, structure-based, and surface-based methods respectively. Scores for the baselines are directly reported from RDE-Net (Luo et al., 2023).

Method	Per-Structure		Overall				
	Pearson	Spearman	Pearson	Spearman	RMSE	MAE	AUROC
Rosetta	0.3284	0.2988	0.3113	0.3468	1.6173	1.1311	0.6562
FoldX	0.3789	0.3693	0.3120	0.4071	1.9080	1.3089	0.6582
ESM-1v	0.0073	-0.0118	0.1921	0.1572	1.9609	1.3683	0.5414
PSSM	0.0826	0.0822	0.0159	0.0666	1.9978	1.3895	0.5260
MSA Transf.	0.1031	0.0868	0.1173	0.1313	1.9835	1.3816	0.5768
Tranception	0.1348	0.1236	0.1141	0.1402	2.0382	1.3883	0.5885
ESM-IF	0.2241	0.2019	0.3194	0.2806	1.8860	1.2857	0.5899
B-factor	0.2042	0.1686	0.2390	0.2625	2.0411	1.4402	0.6044
DDGPred	0.3750	0.3407	0.6580	0.4687	**1.4998**	**1.0821**	0.6992
End-to-End	0.3873	0.3587	0.6373	0.4882	1.6198	1.1761	0.7172
MIF-Net.	0.3965	0.3509	0.6523	0.5134	1.5932	1.1469	0.7329
RDE-Net.	<u>0.4448</u>	<u>0.4010</u>	0.6447	<u>0.5584</u>	<u>1.5799</u>	<u>1.1123</u>	<u>0.7454</u>
PPIFormer	0.4281	0.3995	<u>0.6450</u>	0.5304	1.6420	1.1186	0.7380
Surface-VQMAE	**0.4694**	**0.4324**	**0.6482**	**0.5611**	1.5876	1.1271	**0.7469**

Dataset and Metrics. Evaluation is carried out in the widely recognized SKEMPI.v2 database (Jankauskaitė et al., 2019). It contains data on changes in the thermodynamic parameters and kinetic rate constants after mutation for structurally resolved PPIs. The latest version contains manually curated binding data for 7,085 mutations. The dataset is split into three folds by structure, each containing unique protein complexes that do not appear in other folds. Two folds are used for train and validation, and the remaining one is used for test. This yields three different sets of parameters and ensures that every data point in SKEMPI.v2 is tested once.

We employ five metrics to assess the accuracy of $\Delta\Delta G$ predictions, encompassing Pearson and Spearman correlation coefficients, RMSE, mean absolute error (MAE), and AUROC. Calculating AUROC involves classifying mutations according to the direction of their $\Delta\Delta G$ values. In real-world scenarios, particular attention is given to the correlation observed within a specific protein complex. To address this, we adopt the approach of Luo et al. (2023) by organizing mutations according to their associated structures. Groups with fewer than ten mutation data points are excluded from this analysis. Subsequently, correlation calculations are conducted independently for each structure, introducing two additional metrics: the average per-structure Pearson and Spearman correlation coefficients.

Baselines. We evaluate the effectiveness of Surface-VQMAE against various categories of techniques. The initial kind encompasses conventional empirical energy functions such as **Rossetta** Cartesian $\Delta\Delta G$ (Alford et al., 2017) and **FoldX**. The second grouping comprises sequence or evolution-based methodologies, exemplified by **ESM-1v** (Meier et al., 2021), **PSSM** (position-specific scoring matrix), **MSA Transformer** (Rao et al., 2021), **Tranception** (Notin et al., 2022), and **ESM-IF** (Hsu et al., 2022). The third category includes structure-based models such as **DDGPred** (Shan et al., 2022), **End-to-End**, and Masked Inverse Folding (**MIF-Network**) (Yang et al., 2022) that adopt Graph Transformer (GT) (Luo et al., 2023) as the encoder with an MLP to directly forecast $\Delta\Delta G$. Besides, **B-factors** is the network that anticipates the B-factor of residues and incorporates the projected B-factor in lieu of entropy for $\Delta\Delta G$ prediction. Lastly, Rotamer Density Estimator (**RDE-Network**) (Luo et al., 2023) uses a flow-based generative model to estimate the probability distribution of rotamers and uses entropy to measure flexibility. **PPI-Former** (Bushuiev et al., 2023) is pretrained on a newly collected non-redundant 3D PPI interface dataset PPIRef through the mask language modeling (MLM) technique.

Results. As displayed in Table 3, our Surface-VQMAE is better or more competitive in all regression metrics. Precisely, it achieves the highest per-structure Spearman (0.4324) and Pearson's (0.4694) correlations, which are considered the primary metrics because the correlation of one specific protein complex is the most important. Performance on subsets of single-mutation and multi-mutation is removed to Appendices 5 and 6 due to space limitation. In particular, multiple point mutations are often required for successful affinity maturation (Sulea et al., 2018), and Surface-VQMAE outperforms RDE-Net and PPIFormer by a large margin in the multi-mutation subset.

4.4. Ablation Studies and Visualization

We conduct additional experiments to investigate the contributions of different components of Surface-VQMAE and display the analysis in Appendix C.2. It can be found that

the removal of SurfFormer, the introduction of MAE, and the VQ technique, all induce performance detriment. This phenomenon verifies the necessity of our hierarchical backbone design and the constraint on MAE's latent variable space. Additionally, we visualize the impact of different FPS downsampling ratios and give some examples of surface patches in Appendix C.3.

5. Conclusion

Interactions between proteins and other biomolecules are the basis of protein function in most biological processes. In this study, we introduce a novel surface-based protein structural pretraining method, *i.e.*, Surface-VQMAE, to excavate information from massive unlabeled molecular surfaces. We validate its effectiveness through three vital and challenging downstream tasks, containing protein interface scoring, antibody-antigen binding prediction, and mutant effect prediction.

Limitations

Despite the fruitful progress of Surf-VQMAE in taking the first step for surface-based pretraining, there exists still room for future improvements. A potential limitation arises from solely relying on a single downsampling technique (FPS) without investigating alternative methods. Exploring different approaches could reveal varying trade-offs between representation quality and computational efficiency. Moreover, our study may not thoroughly examine the influence of noise or outliers in the surface data, which could impact the downsampling process and subsequent analysis outcomes. Furthermore, the generalizability of our findings might be constrained to specific protein structures or datasets utilized in the study such as SKEMPI and SAbDab. It underscores the need for caution when extrapolating the proposed approach to diverse contexts.

Acknowledgements

We would like to thank Professor Le Song and Dr. Hui Li at Biomap, Professor Jinbo Xu and Dr. Wuwei Tan at MoleculeMind, Dr. Siyuan Li and Dr. Bozhen Hu at Westlake University for their helpful feedback and discussion. This work was supported by the Science & Technology Innovation 2030 Major Program Project No. 2021ZD0150100, National Natural Science Foundation of China Project No. U21A20427, Project No. WU2022A009 from the Center of Synthetic Biology and Integrated Bioengineering of Westlake University, and Project No. WU2023C019 from the Westlake University Industries of the Future Research.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning for Life Science. There are many potential societal consequences of our work, which be specifically highlighted here. Firstly, by improving the accuracy of protein function predictions, Surface-VQMAE can expedite the identification of potential drug targets and the design of effective pharmaceuticals. This could lead to faster development of treatments for various diseases, including those currently difficult to treat. Secondly, a better understanding of PPIs and functions can contribute to more tailored medical treatments based on individual genetic profiles. This can improve treatment efficacy and reduce adverse effects, ultimately enhancing patient outcomes.

References

Afham, M., Dissanayake, I., Dissanayake, D., Dharmasiri, A., Thilakarathna, K., and Rodrigo, R. Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9902–9912, 2022.

Alford, R. F., Leaver-Fay, A., Jeliazkov, J. R., O'Meara, M. J., DiMaio, F. P., Park, H., Shapovalov, M. V., Renfrew, P. D., Mulligan, V. K., Kappel, K., et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13 (6):3031–3048, 2017.

Andrusier, N., Nussinov, R., and Wolfson, H. J. Firedock: fast interaction refinement in molecular docking. *Proteins: Structure, Function, and Bioinformatics*, 69(1): 139–159, 2007.

Baspinar, A., Cukuroglu, E., Nussinov, R., Keskin, O., and Gursoy, A. Prism: a web server and repository for prediction of protein–protein interactions and modeling their 3d complexes. *Nucleic acids research*, 42(W1):W285–W289, 2014.

Bushuiev, A., Bushuiev, R., Filkin, A., Kouba, P., Gabrielova, M., Gabriel, M., Sedlar, J., Pluskal, T., Damborsky, J., Mazurenko, S., et al. Learning to design protein-protein interactions with enhanced generalization. *arXiv preprint arXiv:2310.18515*, 2023.

Charlier, B., Feydy, J., Glaunes, J. A., Collin, F.-D., and Durif, G. Kernel operations on the gpu, with autodiff, without memory overflows. *The Journal of Machine Learning Research*, 22(1):3457–3462, 2021.

Chen, G., Wang, M., Yang, Y., Yu, K., Yuan, L., and Yue, Y. Pointgpt: Auto-regressively generative pre-training from point clouds. *arXiv preprint arXiv:2305.11487*, 2023.

Cheng, T. M.-K., Blundell, T. L., and Fernandez-Recio, J. pydock: electrostatics and desolvation for effective scoring of rigid-body protein–protein docking. *Proteins: Structure, Function, and Bioinformatics*, 68(2):503–515, 2007.

Choy, C., Gwak, J., and Savarese, S. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3075–3084, 2019.

Connolly, M. L. Analytical molecular surface calculation. *Journal of applied crystallography*, 16(5):548–558, 1983.

Duff, T., Burgess, J., Christensen, P., Hery, C., Kensler, A., Liani, M., and Villemin, R. Building an orthonormal basis, revisited. *JCGT*, 6(1), 2017.

Dunbar, J., Krawczyk, K., Leem, J., Baker, T., Fuchs, A., Georges, G., Shi, J., and Deane, C. M. Sabdab: the structural antibody database. *Nucleic acids research*, 42 (D1):D1140–D1146, 2014.

Fan, H., Su, H., and Guibas, L. J. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 605–613, 2017.

Gainza, P., Sverrisson, F., Monti, F., Rodola, E., Boscaini, D., Bronstein, M., and Correia, B. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. *Nature Methods*, 17(2):184–192, 2020.

Guo, Z. and Yamaguchi, R. Machine learning methods for protein-protein binding affinity prediction in protein design. *Frontiers in Bioinformatics*, 2:1065703, 2022.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Hermosilla, P. and Ropinski, T. Contrastive representation learning for 3d protein structures. *arXiv preprint arXiv:2205.15675*, 2022.

Hsu, C., Verkuil, R., Liu, J., Lin, Z., Hie, B., Sercu, T., Lerer, A., and Rives, A. Learning inverse folding from millions of predicted structures. *bioRxiv*, 2022.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Jankauskaitė, J., Jiménez-García, B., Dapkūnas, J., Fernández-Recio, J., and Moal, I. H. Skempi 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics*, 35(3):462–469, 2019.

Jin, W., Sarkizova, S., Chen, X., Hacohen, N., and Uhler, C. Unsupervised protein-ligand binding energy prediction via neural euler's rotation equation. *arXiv preprint arXiv:2301.10814*, 2023.

Joosten, R. P., Long, F., Murshudov, G. N., and Perrakis, A. The pdb_redo server for macromolecular structure model optimization. *IUCrJ*, 1(4):213–220, 2014.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lee, Y., Yu, H., Lee, J., and Kim, J. Pre-training sequence, structure, and surface features for comprehensive protein representation learning. In *The Twelfth International Conference on Learning Representations*, 2023.

Leem, J., Mitchell, L. S., Farmery, J. H., Barton, J., and Galson, J. D. Deciphering the language of antibodies using self-supervised learning. *Patterns*, pp. 100513, 2022.

Leman, J. K., Weitzner, B. D., Lewis, S. M., Adolf-Bryfogle, J., Alam, N., Alford, R. F., Aprahamian, M., Baker, D., Barlow, K. A., Barth, P., et al. Macromolecular modeling and design in rosetta: recent methods and frameworks. *Nature methods*, 17(7):665–680, 2020.

Li, P. and Liu, Z.-P. Geobind: segmentation of nucleic acid binding interface on protein surface with geometric deep learning. *Nucleic Acids Research*, 51(10):e60–e60, 2023.

Li, S., Zhang, L., Wang, Z., Wu, D., Wu, L., Liu, Z., Xia, J., Tan, C., Liu, Y., Sun, B., et al. Masked modeling for self-supervised representation learning on vision and beyond. *arXiv preprint arXiv:2401.00897*, 2023.

Lin, H., Zhang, O., Zhao, H., Wu, L., Jiang, D., Liu, Z., Huang, Y., and Li, S. Z. Ppflow: Target-aware peptide design with torsional flow matching. *bioRxiv*, pp. 2024–03, 2024.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv*, 2022.

Liu, S., Zhu, T., Ren, M., Yu, C., Bu, D., and Zhang, H. Predicting mutational effects on protein-protein binding via a side-chain diffusion probabilistic model. *arXiv preprint arXiv:2310.19849*, 2023.

Liu, Z., Li, Y., Han, L., Li, J., Liu, J., Zhao, Z., Nie, W., Liu, Y., and Wang, R. Pdb-wide collection of binding data: current status of the pdbbind database. *Bioinformatics*, 31(3):405–412, 2015.

Luo, S., Su, Y., Wu, Z., Su, C., Peng, J., and Ma, J. Rotamer density estimator is an unsupervised learner of the effect of mutations on protein-protein interaction. *bioRxiv*, pp. 2023–02, 2023.

Ma, X., Qin, C., You, H., Ran, H., and Fu, Y. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv preprint arXiv:2202.07123*, 2022.

Meier, J., Rao, R., Verkuil, R., Liu, J., Sercu, T., and Rives, A. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in Neural Information Processing Systems*, 34:29287–29303, 2021.

Mitra, N. J. and Nguyen, A. Estimating surface normals in noisy point cloud data. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pp. 322–328, 2003.

Morton, G. M. A computer oriented geodetic data base and a new technique in file sequencing. 1966.

Mylonas, S. K., Axenopoulos, A., and Daras, P. Deepsurf: a surface-based deep learning approach for the prediction of ligand binding sites on proteins. *Bioinformatics*, 37 (12):1681–1690, 2021.

Myung, Y., Pires, D. E., and Ascher, D. B. Csm-ab: Graph-based antibody–antigen binding affinity prediction and docking scoring function. *Bioinformatics*, 38(4):1141–1143, 2022.

Notin, P., Dias, M., Frazer, J., Hurtado, J. M., Gomez, A. N., Marks, D., and Gal, Y. Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval. In *International Conference on Machine Learning*, pp. 16990–17017. PMLR, 2022.

Pang, Y., Wang, W., Tay, F. E., Liu, W., Tian, Y., and Yuan, L. Masked autoencoders for point cloud self-supervised learning. In *European conference on computer vision*, pp. 604–621. Springer, 2022.

Pierce, B. and Weng, Z. Zrank: reranking protein docking predictions with an optimized energy function. *Proteins: Structure, Function, and Bioinformatics*, 67(4): 1078–1086, 2007.

Pierce, B. and Weng, Z. A combination of rescoring and refinement significantly improves protein docking performance. *Proteins: Structure, Function, and Bioinformatics*, 72(1):270–279, 2008.

Pons, C., Talavera, D., De La Cruz, X., Orozco, M., and Fernandez-Recio, J. Scoring by intermolecular pairwise propensities of exposed residues (sipper): a new efficient potential for protein- protein docking. *Journal of chemical information and modeling*, 51(2):370–377, 2011.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Rao, R. M., Liu, J., Verkuil, R., Meier, J., Canny, J., Abbeel, P., Sercu, T., and Rives, A. Msa transformer. In *International Conference on Machine Learning*, pp. 8844–8856. PMLR, 2021.

Ravikant, D. and Elber, R. Pie—efficient filters and coarse grained potentials for unbound protein–protein docking. *Proteins: Structure, Function, and Bioinformatics*, 78(2): 400–419, 2010.

Riahi, S., Lee, J. H., Sorenson, T., Wei, S., Jager, S., Olfati-Saber, R., Zhou, Y., Park, A., Wendt, M., Minoux, H., et al. Surface id: a geometry-aware system for protein molecular surface comparison. *Bioinformatics*, 39(4): btad196, 2023.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

Sanner, M. F., Olson, A. J., and Spehner, J.-C. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers*, 38(3):305–320, 1996.

Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A., and Müller, K.-R. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24), 2018.

Shan, S., Luo, S., Yang, Z., Hong, J., Su, Y., Ding, F., Fu, L., Li, C., Chen, P., Ma, J., et al. Deep learning guided optimization of human antibody against sars-cov-2 variants with broad neutralization. *Proceedings of the National Academy of Sciences*, 119(11):e2122954119, 2022.

Shaw, P., Uszkoreit, J., and Vaswani, A. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018.

Somnath, V. R., Bunne, C., and Krause, A. Multi-scale representation learning on proteins. *Advances in Neural Information Processing Systems*, 34:25244–25255, 2021.

Stebliankin, V., Shirali, A., Baral, P., Shi, J., Chapagain, P., Mathee, K., and Narasimhan, G. Evaluating protein binding interfaces with transformer networks. *Nature Machine Intelligence*, 5(9):1042–1053, 2023.

Sulea, T., Hussack, G., Ryan, S., Tanha, J., and Purisima, E. O. Application of assisted design of antibody and protein therapeutics (adapt) improves efficacy of a clostridium difficile toxin a single-domain antibody. *Scientific reports*, 8(1):2260, 2018.

Sverrisson, F., Feydy, J., Correia, B. E., and Bronstein, M. M. Fast end-to-end learning on protein surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15272–15281, 2021.

Tian, X., Ran, H., Wang, Y., and Zhao, H. Geomae: Masked geometric target prediction for self-supervised point cloud pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13570–13580, 2023.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Viswanath, S., Ravikant, D., and Elber, R. Improving ranking of models for protein complexes with side chain modeling and atomic potentials. *Proteins: Structure, Function, and Bioinformatics*, 81(4):592–606, 2013.

Vreven, T., Moal, I. H., Vangone, A., Pierce, B. G., Kastritis, P. L., Torchala, M., Chaleil, R., Jiménez-García, B., Bates, P. A., Fernandez-Recio, J., et al. Updates to the integrated protein–protein interaction benchmarks: docking benchmark version 5 and affinity benchmark version 2. *Journal of molecular biology*, 427(19):3031–3041, 2015.

Wu, F., Jin, S., Jiang, Y., Jin, X., Tang, B., Niu, Z., Liu, X., Zhang, Q., Zeng, X., and Li, S. Z. Pre-training of equivariant graph matching networks with conformation flexibility for drug binding. *Advanced Science*, 9(33): 2203796, 2022a.

Wu, F., Tao, Y., Radev, D., and Xu, J. When geometric deep learning meets pretrained protein language models. *arXiv preprint arXiv:2212.03447*, 2022b.

Wu, F., Courty, N., Jin, S., and Li, S. Z. Improving molecular representation learning with metric learning-enhanced optimal transport. *Patterns*, 4(4), 2023a.

Wu, F., Qin, H., Gao, W., Li, S., Coley, C. W., Li, S. Z., Zhan, X., and Xu, J. Instructbio: A large-scale semi-supervised learning paradigm for biochemical problems. *arXiv preprint arXiv:2304.03906*, 2023b.

Wu, F., Radev, D., and Li, S. Z. Molformer: Motif-based transformer on 3d heterogeneous molecular graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 5312–5320, 2023c.

Wu, F., Wu, L., Radev, D., Xu, J., and Li, S. Z. Integration of pre-trained protein language models into geometric deep learning networks. *Communications Biology*, 6(1): 876, 2023d.

Yang, K. K., Zanichelli, N., and Yeh, H. Masked inverse folding with sequence transfer for protein representation learning. *bioRxiv*, pp. 2022–05, 2022.

Yang, Y., Feng, C., Shen, Y., and Tian, D. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 206–215, 2018.

Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., and Lu, J. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19313–19322, 2022.

Zhang, R., Guo, Z., Gao, P., Fang, R., Zhao, B., Wang, D., Qiao, Y., and Li, H. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *Advances in neural information processing systems*, 35: 27061–27074, 2022a.

Zhang, Z., Girdhar, R., Joulin, A., and Misra, I. Self-supervised pretraining of 3d features on any point-cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10252–10263, 2021.

Zhang, Z., Xu, M., Jamasb, A., Chenthamarakshan, V., Lozano, A., Das, P., and Tang, J. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*, 2022b.

Zhang, Z., Xu, M., Lozano, A., Chenthamarakshan, V., Das, P., and Tang, J. Physics-inspired protein encoder pretraining via siamese sequence-structure diffusion trajectory prediction. *arXiv preprint arXiv:2301.12068*, 2023.

A. Analysis of MAE Target

Curvatures. An intuitive drawback of point cloud coordinate reconstruction is that it does not satisfy the roto-translation equivariance using the prevailing scheme (Pang et al., 2022). To overcome this, we propose to leverage surface geometric properties such as curvatures as the unsupervised prediction target. For surfaces (and, more generally for higher-dimensional manifolds) that are embedded in Euclidean space, the concept of curvature depends on the choice of a direction on the surface or manifold. This leads to the concepts of maximal curvature, minimal curvature, and mean curvature. The maximum and minimum normal curvatures at a point on a surface are called the principal (normal) curvatures, and the directions in which these normal curvatures occur are called the principal directions. We recommend interested readers to take a look at this introduction [1] for more details. Here, we verify that curvatures are invariant to 3D rotation and translations. More formally, for any translation vector $o \in \mathbb{R}^3$ and for any orthogonal matrix $Q \in \mathbb{R}^{3\times 3}$, $\boldsymbol{\psi} \in \mathbb{R}^3$ remains the same. Towards this end, we want to show that eigenvalues $\boldsymbol{\epsilon}$ of the co-variance matrix $\boldsymbol{\Sigma}$ are invariant to o and Q. Notably, there are several popular ways to decompose $\boldsymbol{\Sigma}$ such as the singular value decomposition (SVD) and the eigenvalue decomposition (EVD), and we begin with EVD for illustration. After the spatial transformations, the co-variance matrix is obtained as:

$$
\begin{aligned}
\boldsymbol{\Sigma}' &= \frac{1}{k} \sum_{\mathbf{x}_{i,j}^{\mathrm{p,m}} \in \mathbf{X}_i^{\mathrm{p,m}}} \left(Q\mathbf{x}_{i,j}^{\mathrm{p,m}} + o\right)\left(Q\mathbf{x}_{i,j}^{\mathrm{p,m}} + o\right)^\top - \left(Q\mathbf{x}_{i}^{\mathrm{c,m}} + o\right)\left(Q\mathbf{x}_{i}^{\mathrm{c,m}} + o\right)^\top \\
&= \frac{1}{k} \sum_{\mathbf{x}_{i,j}^{\mathrm{p,m}} \in \mathbf{X}_i^{\mathrm{p,m}}} \left(Q\mathbf{x}_{i,j}^{\mathrm{p,m}}\mathbf{x}_{i,j}^{\mathrm{p,m}\top}Q^\top + o\mathbf{x}_{i,j}^{\mathrm{p,m}\top}Q^\top + Q\mathbf{x}_{i,j}^{\mathrm{p,m}}o^\top + oo^\top\right) \\
&\qquad - \left(Q\mathbf{x}_{i}^{\mathrm{c,m}}\mathbf{x}_{i}^{\mathrm{c,m}\top}Q^\top + o\mathbf{x}_{i}^{\mathrm{c,m}\top}Q^\top + Q\mathbf{x}_{i}^{\mathrm{c,m}}o^\top + oo^\top\right)^\top \\
&= Q\boldsymbol{\Sigma}Q^\top + \frac{1}{k} \sum_{\mathbf{x}_{i,j}^{\mathrm{p,m}} \in \mathbf{X}_i^{\mathrm{p,m}}} \left(o\mathbf{x}_{i,j}^{\mathrm{p,m}\top}Q^\top + Q\mathbf{x}_{i,j}^{\mathrm{p,m}}o^\top\right) - o\mathbf{x}_{i}^{\mathrm{c,m}\top}Q^\top - Q\mathbf{x}_{i}^{\mathrm{c,m}}o^\top \\
&= Q\boldsymbol{\Sigma}Q^\top + o\left(\frac{1}{k} \sum_{\mathbf{x}_{i,j}^{\mathrm{p,m}} \in \mathbf{X}_i^{\mathrm{p,m}}} \mathbf{x}_{i,j}^{\mathrm{p,m}\top}\right)Q^\top + Q\left(\frac{1}{k} \sum_{\mathbf{x}_{i,j}^{\mathrm{p,m}} \in \mathbf{X}_i^{\mathrm{p,m}}} \mathbf{x}_{i,j}^{\mathrm{p,m}}\right)o^\top - o\mathbf{x}_{i}^{\mathrm{c,m}\top}Q^\top - Q\mathbf{x}_{i}^{\mathrm{c,m}}o^\top \\
&= Q\boldsymbol{\Sigma}Q^\top.
\end{aligned}
\tag{19}
$$

Then we further prove that $\boldsymbol{\Sigma}'$ shares the same eigenvalues $\boldsymbol{\epsilon}$ with $\boldsymbol{\Sigma}$ but with transformed eigenvectors. Explicitly, suppose $\{\boldsymbol{\Upsilon}_i\}_{i=1}^3$ are the original eigenvectors of $\boldsymbol{\Sigma}$, then we can get:

$$
\boldsymbol{\Sigma}'\left(Q\boldsymbol{\Upsilon}_iQ^\top\right) = Q\boldsymbol{\Sigma}Q^\top Q\boldsymbol{\Upsilon}_iQ^\top = Q\boldsymbol{\Sigma}\mathbf{I}\boldsymbol{\Upsilon}_iQ^\top = \epsilon_i\left(Q\boldsymbol{\Upsilon}_iQ^\top\right). \tag{20}
$$

In other words, $Q\boldsymbol{\Upsilon}_iQ^\top$ becomes the eigenvector of $\boldsymbol{\Sigma}$ in terms of the corresponding eigenvalue ϵ_i. As a consequence, curvatures $\boldsymbol{\psi}$, which entirely depends on $\boldsymbol{\epsilon}$, will not change regardless of any o and Q. Moreover, due to the relationship of singular values and eigenvalues, *i.e.*, $\xi_i = \sqrt{\epsilon_i}$, curvatures that are computed based on SVD are also not affected by transformations on 3D coordinate systems.

B. Experimental Details

We implement all experiments on 4 A100 GPUs, each with 80G memory. During the pretraining stage, Surface-VQMAE is trained with an Adam optimizer (Kingma & Ba, 2014) with a weight decay of $5.e-3$ and with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. A ReduceLROnPlateau scheduler is employed to automatically adjust the learning rate with a patience of 5 epochs and a minimum learning rate of $1.e-7$. The batch size is set to 32 and an initial learning rate is $1.e-4$. The maximum iterations are 200K with warmingup iterations of 10K and the validation frequency is 1K iterations. The random seed is fixed as 2023. Moreover, we empirically calculate the overlap ratio of all patches and discover a pretty low score of 5.34%.

Hyperparameter Search Space. At the beginning, we adopt a random search to find the best combination of hyperparameters for the backbone architecture (*i.e.*, the point cloud network, and SurfFormer) in three different downstream tasks with only supervised learning. Then we fix these subsets of hyperparameters to build three backbone architectures

[1] A quick and dirty introduction to the curvature of surfaces: http://wordpress.discretization.de/geometryprocessingandapplicationsws19/a-quick-and-dirty-introduction-to-the-curvature-of-surfaces/

and further explore the hyperparameters for the VQMAE-stype pretraining. The details of the hyperparameter setup of Surface-VQMAE are listed in Table 4.

Table 4. Hyperparameters setup for Surface-VQMAE

Hyperparameters Search Space	Symbol	Value
Surface Generation and Partition		
Upsampling Ratio	η	[10, 20]
Surface Radius	r	[1.05]
FPS Downsampling Ratio	ρ	[0.01, 0.02, 0.05, 0.1]
Number of Nearest Point in a Patch	K	[10, 25, 50, 100]
Backbone Achitecture		
Dimension of Point Chemical Features in GeoAN	ϕ_h	[4, 8, 16]
Dimension of Residue Type Embedding in GeoAN	ϕ_R	[16]
Dimension of Patch Chemical Features in SurfFormer	ϕ_{p}	[16, 64, 128]
Number of Nearest Residues in GeoAN	ζ	[8, 16]
Gaussian Kernel Size	$\sigma_{\mathbf{n}}$	[9Å, 12Å]
Radius in the Filters	σ_{d}	[9Å, 12Å]
Scaler Coefficient in Radial Basis Function	γ	[5, 10]
Layer Number of Point Cloud Networks	L_1	[1, 2, 3]
Layer Number of SurfFormer	L_2	[1, 3, 6]
Number of Heads in SurfFormer	–	[4, 8]
Dropout Rate in Point Cloud Network	–	[0.0, 0.1]
Dropout Rate in SurfFormer	–	[0.0, 0.1]
VQMAE Setup		
Masking Ratio	δ	[30%, 50%, 60%, 70%, 80%]
Codebook Size	N_B	[100, 1000]
Target Temperature	τ	[0.0625]
Reconstruction Loss Weight	ν_1	[1.0]
Curvature Loss Weight	ν_2	[0.5, 1.0]
KL-divergence Loss Weight	ν_3	[0.1, 0.5, 1.0]
Training Setup		
Batch Size	–	[16, 32]
Initial Learning Rate	–	[5e-4, 1e-4, 5e-5, 1e-6]
Number of Warmup Iterations	–	[5K, 10K]

C. Additional Results

C.1. Performance on Subsets of SKEMPI.v2

We explicitly document the evaluation results of different methods on the multi-mutation and single-mutation subsets of the SKEMPI.v2 dataset in Table 5 and Table 6. It can be discovered that Surface-VQMAE achieves the best per-structure metrics on both multi-mutation and single-mutation subsets. This indicates that Surface-VQMAE is a more effective tool to screen and select mutant proteins for desired properties.

C.2. Ablation Studies

We investigate the contribution of each component of Surface-VQMAE. To be specific, we take the vanilla point cloud network as the base model (*i.e.*, line 1 in Table 7) and compare it with different settings. As illustrated in Table 7, the removal of each part leads to a significant decline in metrics for different PPI applications. Specifically, we find the surface property objective brings more improvements than the point statistics objective, illustrating the importance of high-level geometric pretext tasks.

Table 5. Evaluation of mutant effect prediction ($\Delta\Delta G$) on the multi-mutation subset of the SKEMPI.v2 dataset.

Method	Per-Structure		Overall				
	Pearson	Spearman	Pearson	Spearman	RMSE	MAE	AUROC
Rosetta	0.1915	0.0836	0.1991	0.2303	2.6581	2.0246	0.6207
FoldX	0.3908	0.3640	0.3560	0.3511	1.5576	1.0713	0.6478
ESM-1v	-0.0599	-0.1284	0.1923	0.1749	2.7586	2.1193	0.5415
PSSM	-0.0174	-0.0504	-0.1126	-0.0458	2.7937	2.1499	0.4442
MSA Transf.	-0.0097	-0.0400	0.0067	0.0030	2.8115	2.1591	0.4870
Tranception	-0.0688	-0.0120	-0.0185	-0.0184	2.9280	2.2359	0.4874
ESM-IF	0.2016	0.1491	0.3260	0.3353	2.6446	1.9555	0.6373
B-factor	0.2078	0.1850	0.2009	0.2445	2.6557	2.0186	0.5876
DDGPred	0.3912	0.3896	0.5938	0.5150	2.1813	1.6699	0.7590
End-to-End	0.4178	0.4034	0.5858	0.4942	2.1971	1.7087	0.7532
MIF-Net.	0.3968	0.3789	0.6139	0.5370	2.1399	1.6422	0.7735
RDE-Net.	0.4233	0.3926	0.6288	0.5900	**2.0980**	**1.5747**	0.7749
PPIFormer	0.3985	0.3925	**0.6405**	**0.5946**	2.1407	1.5753	**0.7893**
Surface-VQMAE	**0.4593**	**0.4444**	0.6242	0.5870	2.1295	1.6139	0.7852

Table 6. Evaluation of mutant effect prediction ($\Delta\Delta G$) on the single-mutation subset of the SKEMPI.v2 dataset.

Method	Per-Structure		Overall				
	Pearson	Spearman	Pearson	Spearman	RMSE	MAE	AUROC
Rosetta	0.3284	0.2988	0.3113	0.3468	1.6173	1.1311	0.6562
FoldX	0.3908	0.3640	0.3560	0.3511	1.5576	1.0713	0.6478
ESM-1v	0.0422	0.0273	0.1914	0.1572	1.7226	1.1917	0.5492
PSSM	0.1215	0.1229	0.1224	0.0997	1.7420	1.2055	0.5659
MSA Transf.	0.1415	0.1293	0.1755	0.1749	1.7294	1.1942	0.5917
Tranception	0.1912	0.1816	0.1871	0.1987	1.7455	1.1708	0.6089
ESM-IF	0.2308	0.2090	0.2957	0.2866	1.6728	1.1372	0.6051
B-factor	0.1884	0.1661	0.1748	0.2054	1.7242	1.1889	0.6100
DDGPred	0.3711	0.3427	0.6515	0.4390	1.3285	0.9618	0.6858
End-to-End	0.3818	0.3426	0.6605	0.4594	1.3148	0.9569	0.7019
MIF-Net.	0.3952	0.3479	**0.6667**	0.4802	1.3052	0.9411	0.7175
RDE-Net.	0.4687	0.4333	0.6421	**0.5271**	1.3333	0.9392	0.7367
PPIFormer	0.4192	0.3796	0.6287	0.4772	1.4232	0.9562	0.7213
Refine-PPI	**0.4751**	**0.4410**	0.6619	0.5180	**1.3326**	**0.9362**	**0.7368**

C.3. Visualization of Surface Patches

We visualize the downsampled point clouds under different FPS downsampling ratios in Figure 2. It was found that as the downsampling ratio decreases, the patch center points become more sparse. However, those center points are evenly allocated and roughly describe the overall shape of the entire protein point cloud. Moreover, we give two specific examples in Figure 3 to illustrate the overlap and distribution of surface patches.

Table 7. Ablation study of Surface-VQMAE, where BSS, BAP, and MEP are the abbreviation of binding site scoring, binding affinity prediction, and mutation effect prediction, respectively.

	SurfFormer	MAE		VQ	BSS	BAP	MEP	
		Point Stat.	Surface Prop.		AUROC	Pearson	Pearson	Spearman
1	✗	✗	✗	✗	89.71	0.318	0.4175	0.3902
2	✓	✗	✗	✗	92.44	0.356	0.4388	0.4026
3	✓	✓	✗	✗	92.68	0.361	0.4397	0.4051
4	✓	✓	✓	✗	94.15	0.397	0.4504	0.4265
5	✓	✓	✓	✓	**94.78**	**0.411**	**0.4694**	**0.4324**

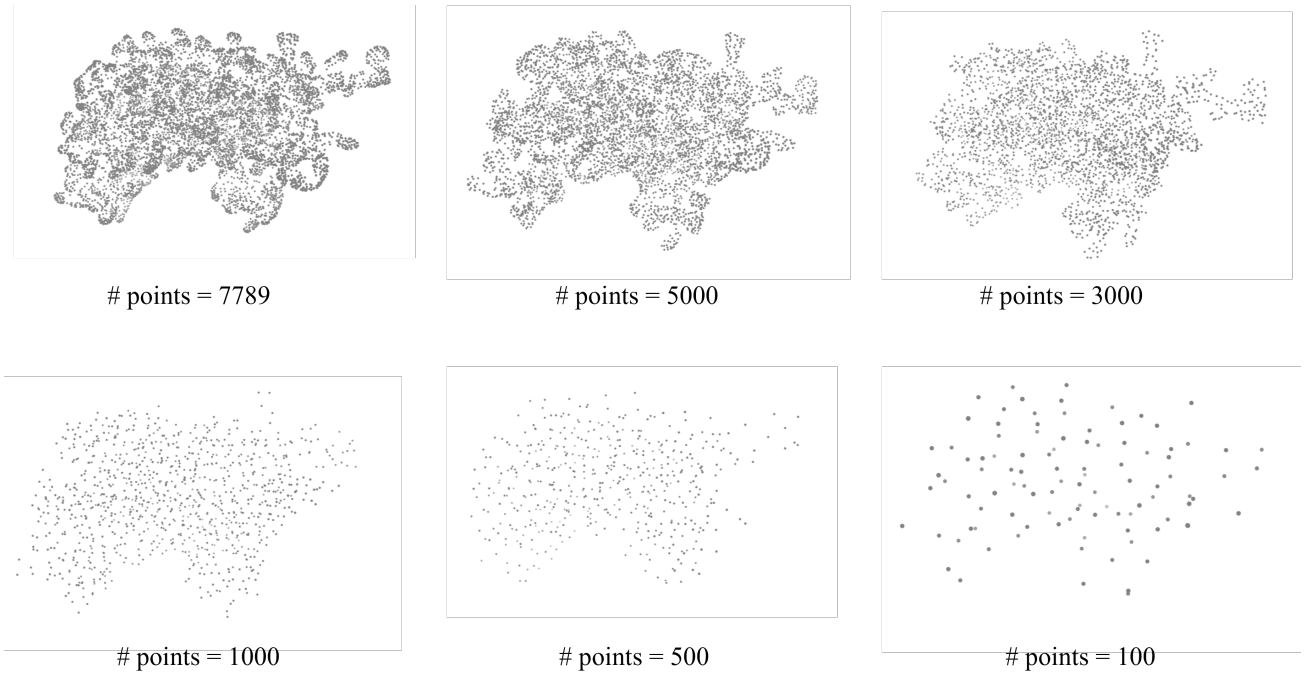

Figure 2. Plots of protein surfaces with different FPS downsampling ratios.

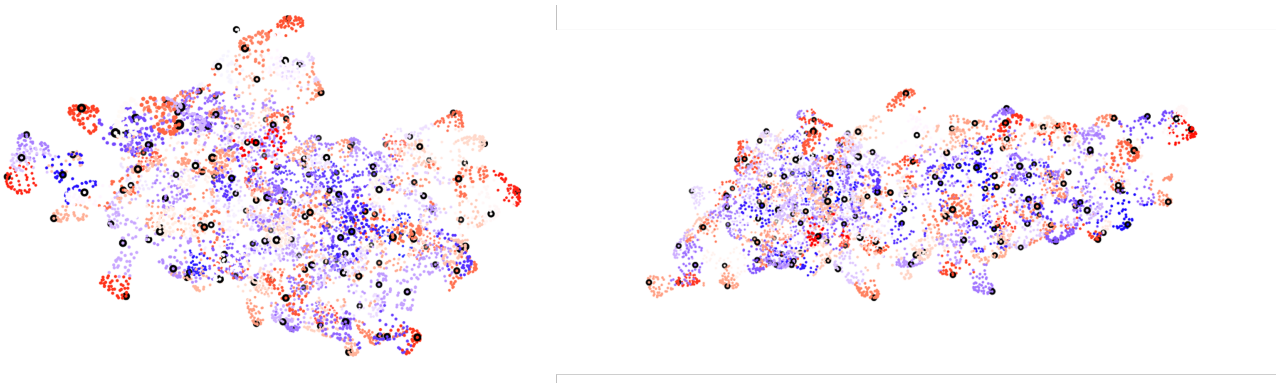

Figure 3. Plots of surface patches in two example proteins.