Point Cloud Based Deep Molecular Pose Estimation for Structure-Based Virtual Screening

Ken Kariya[1][0000−0002−8981−7046], Go Irie[1][0000−0002−4309−4700], Ryosuke Furuta[2][0000−0003−1441−889X], Yota Yamamoto[1][0000−0002−1679−5050], Shin Aoki[1][0000−0002−4287−6487], and Yukinobu Taniguchi[1][0000−0003−3290−1041]

[1] Tokyo University of Science, Tokyo, Japan
4621508@ed.tus.ac.jp, goirie@ieee.org, {yy-yamamoto, shinaoki, taniguchi.yukinobu}@rs.tus.ac.jp
[2] The University of Tokyo, Tokyo, Japan
furuta@iis.u tokyo.ac.jp

Abstract. Structure-Based Virtual Screening (SBVS) is a computer-based simulation technique to streamline the drug design process by identifying candidate inhibitor structures that are likely to bind to a target protein based on the shape similarity of molecular structures. Specifically, the molecular surface is acquired as a point cloud and then the point cloud alignment method is applied; this method continues to be actively studied in computer vision. Since the protein binding site and the inhibitor are not the same objects, their shapes do not match perfectly with numerous gaps between them. Therefore, conventional point cloud alignment methods may provide unsatisfactory performance. This paper proposes an SBVS method based on shape features. Specifically, the proposed method comprises (i) docking simulation based on a learning-based alignment model that simultaneously estimates pose and expansion parameters and (ii) scoring by Truncated Chamfer Distance with expansion transformation. Experiments show that the proposed method yields faster and more accurate SBVS processing than previous methods using optimization of chemical properties.

Keywords: Inhibitor Retrieval · Structure-Based Virtual Screening · Point Cloud · Point Cloud Alignment

1 Introduction

Many diseases are triggered by the binding of a specific substrate (ligand) to an enzyme protein, causing a harmful chemical reaction. Drugs work by pre-emptively bind another ligand (inhibitor) to the enzyme protein to suppress the harmful chemical reaction. Therefore, identifying inhibitors that are likely to bind to a target enzyme protein is critical in the drug design process. However, conducting chemical experiments to identify candidate inhibitors from the huge library of possible ligands is highly inefficient. The experimental confirmation of binding to proteins is labor-intensive and expensive. To reduce this labor and

cost, computer-based simulations (virtual screening) is widely used. The goal of virtual screening is to streamline ligand identification through computer-based simulations.

There are two major types of virtual screening [12]. Ligand-Based Virtual Screening (LBVS) [15, 27, 6] and Structure-Based Virtual Screening (SBVS) [25, 1, 11]. LBVS searches for inhibitors that are similar to known ligands that bind to the target enzyme protein, whereas SBVS evaluates the likelihood of binding by the molecular structural similarity between the target enzyme protein and the ligands. In this paper, we focus on SBVS, which can be applied without prior knowledge of which ligands bind to the target enzyme protein.

There are several SBVS methods [16]. DOCK [1] has the longest history and uses scores based on physicochemical values such as van der Waals forces. In addition, empirical weighted scores and systematic search algorithms that avoid trying to search solutions in spaces known to lead to the wrong solution have been developed and used in Glide [11]. Recently, AutoDock Vina [25], which use an even more improved score function and a genetic algorithm, has become one of the most widely used methods. These methods compute scores for a large number of randomly generated poses during structure exploration, but the calculation time needed is excessive. In addition, shape similarity is evaluated using chemical properties such as van der Waals forces. However, using chemical properties may negatively affect the score, and examples include slight collisions between the inhibitor and the pocket that may cause score errors [26].

In this paper, we apply the point cloud alignment method, actively studied in computer vision, to SBVS to reduce computation time and to better evaluate shape similarity. We represent the molecular surfaces of the pocket (binding site) and the candidate inhibitor as a point cloud and align them by predicting the binding pose. Then we evaluate shape similarity by calculating the point cloud displacement needed to attain the predicted pose. To the best of our knowledge, this is the first proposed method for SBVS based only on shape from docking to scoring using point clouds. The proposed method can find the pose in which the inhibitor can fit into the pocket faster and achieve a higher accuracy in virtual screening. Since point cloud alignment has been extensively studied in recent years, we choose the point cloud as the shape representation.

There are two differences between the general point cloud alignment problem and point cloud alignment in SBVS (Fig. 2). The first difference is that SBVS alignment targets different objects, not the same object adopted in general point cloud alignment. The second difference is that gaps occur even when the inhibitor and pocket are bound in the correct position. This is because not only geometrical similarity but also chemical binding forces such as electrostatic forces contribute to the binding of the inhibitor and the protein. To adapt the point cloud-based approach to SBVS, we propose (i) a learning-based point cloud alignment method that simultaneously estimates the pose at the merge and the amount of expansion needed to fill the gap, and (ii) a scoring method using Truncated Chamfer Distance with expansion transformation. Experiments

on the DUD-E dataset [18] to evaluate virtual screening yield analytical results showing that the proposal is effective and superior to existing SBVS methods.

2 Related Work

2.1 Structure-Based Virtual Screening

Fig. 1 shows the processing pipeline of SBVS. (i) docking simulation: specify a query protein pocket and a candidate inhibitor, then predict the pose when the inhibitor binds to the pocket; (ii) scoring: calculate a score representing the stability of the predicted binding state; (iii) ranking: order the inhibitors based on their scores.

DOCK [1], Glide [11], and AutoDock Vina [25] are all methods that predict binding by solving a chemical energy minimization problem focusing on chemical

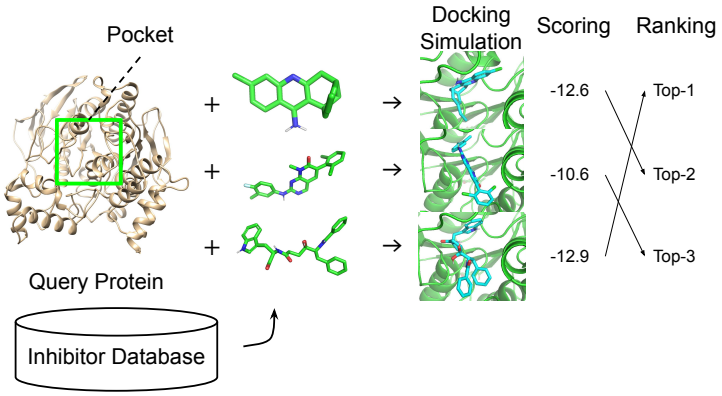
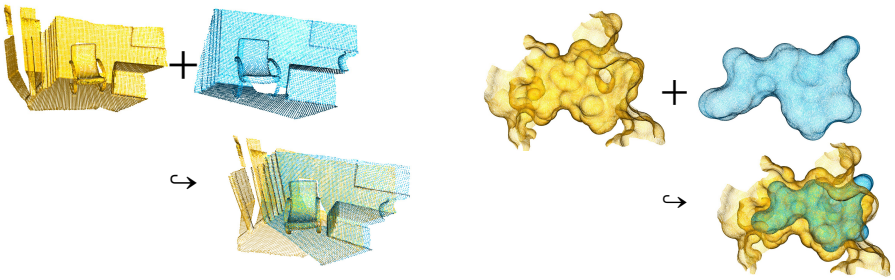


Fig. 1: Illustration of Structure-Based Virtual Screening (SBVS) method



(a) General point cloud alignment [29] **(b)** alignment of pocket and inhibitor

Fig. 2: Comparison of point cloud alignments

affinity. In the latest benchmark CASF-2016 [24], which is a comparative evaluation of scoring capabilities, AutoDock Vina obtained the highest docking power among all docking tools. In recent years, several methods using deep learning have been proposed. EquiBind [23] applies graph neural networks to docking. GNINA [17] scores the binding stabilities using deep learning. DiffDock [7] uses a diffusion generative model for docking.

Since AutoDock Vina was improved in 2021 [9] and has a successful track record in developing clinically approved HIV-1 integrase inhibitors [13], it is used to benchmark the proposed method.

2.2 Point Cloud Alignment / Pose Estimation

The proposed method in this paper uses point cloud alignment to determine docking. Point cloud alignment creates a rigid transformation from one point cloud to the other with the goal of better fitting the two point clouds. Here, we discuss related methods for point cloud alignment.

Iterative Closest Point (ICP) [3] is one of the most common methods for point cloud alignment, but it has a problem in its dependence on the initial relative poses of the point clouds. As a method that does not depend on initial values, the Fast Point Feature Histograms (FPFH) [21] uses features to describe each point's local characteristics, and the points in the two point clouds with similar FPFH features are registered as corresponding points. Then, the point clouds are aligned so that the corresponding points are closest to each other. Here, finding the corresponding points is called registration.

PointNet [19] is a representative neural network model for point cloud processing and has shown to be helpful in segmentation and classification tasks. Guo et al. [14] proposed the Point Cloud Transformer (PCT) based on Transformer, which has shown great success in natural language processing and great potential in image processing. In addition, they proposed an input embedding method that performs feature aggregation of neighboring points by farthest point sampling and nearest neighbor search to better capture the local context within a point cloud. PCT achieves state-of-the-art performance on shape classification, part segmentation, semantic segmentation, and the estimation of normals tasks. Several extent works apply learned 3D descriptors for point cloud alignment. Aoki et al. [2] proposed a deep learning model PointNetLK that solves the alignment problem by minimizing the distance between the fixed-length global descriptors generated by PointNet. Sarode et al. [22] proposed a deep learning model, PCRNet, which similarly uses PointNet to extract features and a Fully Connected (FC) layer to solve the alignment problem. We use PointNetLK and PCRNet as baseline models and compare their accuracy with that of the alignment model proposed in this paper.

Some studies have adapted point cloud alignment to chemical pose estimation. Douguet et al. [8] proposed a point cloud method for comparing shapes and partial shapes between molecules. The van der Waals surface is represented as a point cloud, and point cloud alignment is performed by a registration-based

method using FPFH features, after which the alignment is improved by optimizing the matching of the colored points. This study suggests the possibility of assessing molecular shape similarity. Eguida et al. [10] proposed fragment-based virtual screening using 3D point clouds. Fragment-based virtual screening is a method in which inhibitors are created from small blocks of fragmented molecules called "fragments." Alignment by a registration-based method using FPFH features enables the pockets to be filled with fragments. Their study suggested that point cloud-based computer vision approaches to the protein-ligand docking problem can be developed. In this paper, we adopt a registration-based method that uses the FPFH features of those works for docking and compare its accuracy with the proposed method.

3 Proposed Method

We propose a shape-based SBVS method shown in Fig.3. First, point clouds of pockets and candidate inhibitors are generated. Next, virtual screening is performed in three steps as in general SBVS: (i) docking simulation, (ii) scoring, and (iii) ranking. In this paper, in step (i), docking is treated as an alignment problem to quickly find the pose in which the inhibitor fits into the pocket. Unlike the usual point cloud alignment, there is a certain gap between the point clouds of the inhibitor and the pocket, and the shapes do not perfectly match. To solve the problem, we propose a learning-based alignment model that simultaneously estimates the pose and expansion parameters. In step (ii), the error score between the pocket and the inhibitor is calculated to evaluate how well the estimated pose of the inhibitor fits into the pocket. To better utilize the expansion obtained from docking, we propose a scoring method that uses truncated chamfer distance with expansion transformation.

3.1 Deep Learning-Based Point Cloud Alignment

As shown in Fig. 2b, because the pockets and inhibitor shapes do not perfectly match, registration-based alignment using FPFH, as used by Douguet et al. [8] and Eguida et al. [10], does not provide high accuracy. In this paper, we use a learning-based alignment model to solve this problem.

Let inhibitor surface point cloud be $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots\}$ ($\boldsymbol{x}_i \in \mathbb{R}^3$), and the pocket surface point cloud be $Y = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots\}$ ($\boldsymbol{y}_i \in \mathbb{R}^3$). The architecture of the proposed alignment model is shown in Fig.4. The model takes the inhibitor point cloud X and pocket point cloud Y as input, and outputs the pose parameter $\boldsymbol{z}_{\text{pose}} \in \mathbb{R}^7$, which consists of rotation quaternion $\boldsymbol{q} \in \mathbb{R}^4$ and translation vector $\boldsymbol{t} \in \mathbb{R}^3$, and the expansion parameter $a \in \mathbb{R}$.

Input Embedding. We convert the input point clouds to 256-dimentional features using the input embedding module proposed by Guo et al. in PCT [14]. The module reduces the size of inhibitor point cloud X and pocket point cloud Y to 256 points, and creates a new point cloud X', Y'. The module extracts, simultaneously, features $\mathbf{f}_x \in \mathbb{R}^{256\times|X'|}$ and $\mathbf{f}_y \in \mathbb{R}^{256\times|Y'|}$.

Self-Attention. We use the self-attention layer proposed in PCT to capture global feature. When the input is $\mathbf{f}_{\text{in}} \in \mathbb{R}^{256\times N}$, the self-attention layer outputs $\mathbf{f}_{\text{out}} = \text{LBR}(\text{Scale}(\text{Softmax}(\mathbf{QK}^T)\mathbf{V})) + \mathbf{f}_{\text{in}} \in \mathbb{R}^{256\times N}$ that captures the context of the entire point cloud. N is the number of points, LBR is Linear, BatchNorm, and ReLU layer, Scale is the normalization by $l1$-norm for the second dimension, $\mathbf{Q} \in \mathbb{R}^{64\times N}, \mathbf{K} \in \mathbb{R}^{64\times N}, \mathbf{V} \in \mathbb{R}^{256\times N}$ are the query, key and value matrices, respectively. The four Self-Attention layers yield features $\mathbf{f}_x'^i \in \mathbb{R}^{256\times|X'|}$ and

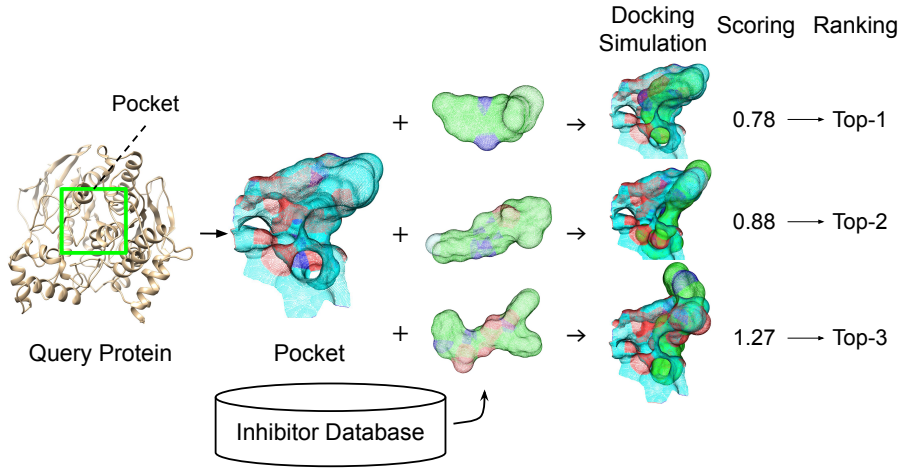


Fig. 3: Illustration of SBVS

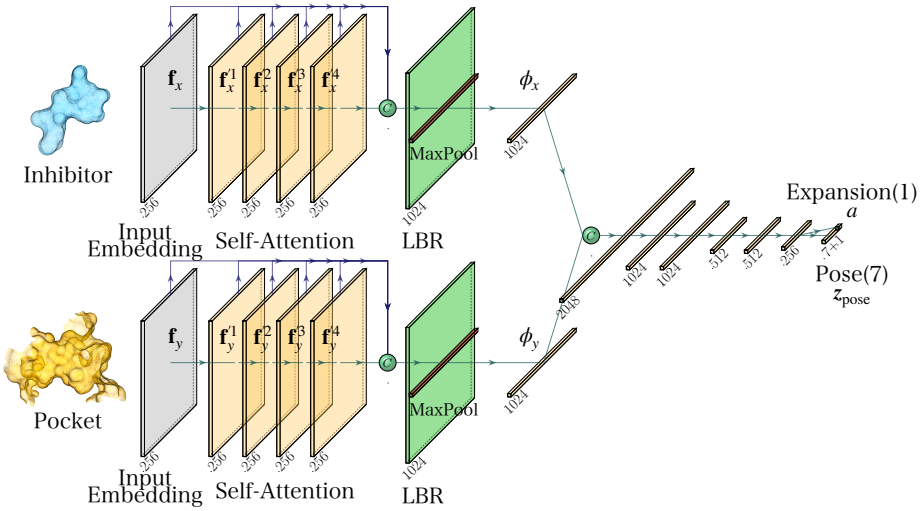


Fig. 4: Proposed alignment model

$\mathbf{f}_y'^i \in \mathbb{R}^{256\times|Y'|}, i = 1,2,3,4$ from features $\mathbf{f}_x$ and $\mathbf{f}_y$. Since the pocket point cloud and the candidate inhibitor point cloud are different objects, we learn different network parameters (weights) in the self-attention layers for pockets and inhibitors although the network parameters in the input embedding module are shared between two streams.

Pose and Expansion Parameter Prediction. Similar to PCT, we concatenate the 256-dimensional local features yielded by the input embedding module and the 256-dimensional global features yielded by each layer of the four self-attention layers. As a result, the dimension of each concatenated feature increases to 1,280. The concatenated features $\mathrm{concat}(\mathbf{f}_x, \mathbf{f}_x'^1, \mathbf{f}_x'^2, \mathbf{f}_x'^3, \mathbf{f}_x'^4) \in \mathbb{R}^{1280\times|X'|}$ and $\mathrm{concat}(\mathbf{f}_y, \mathbf{f}_y'^1, \mathbf{f}_y'^2, \mathbf{f}_y'^3, \mathbf{f}_y'^4) \in \mathbb{R}^{1280\times|X'|}$ are passed through the LBR layer and max pooling, and then we extract two global feature vectors $\boldsymbol{\phi}_x \in \mathbb{R}^{1024}$ and $\boldsymbol{\phi}_y \in \mathbb{R}^{1024}$. $\boldsymbol{\phi}_x$ and $\boldsymbol{\phi}_y$ are input to the FC layers to predict pose parameter $\boldsymbol{z}_{\mathrm{pose}}$ and expansion parameter a. Here, the weights of LBR layer are shared between two streams.

3.2 Scoring

In order to rank the candidate inhibitors for virtual screening, we need to define a score to evaluate how well the inhibitor with the estimated pose fits the pocket. However, the gap between the inhibitor and the pocket has a harmful effect when calculating the score. To solve this problem, we expand the inhibitor to narrow the gap before calculating the score.

Expansion of Molecular Surface Representation. We estimate the expansion of the inhibitor and allow recourse to the general point cloud alignment problem. The molecular surface points $\boldsymbol{x}_i \in \mathbb{R}^3$ are expanded in the normal direction to yield new molecular surface points

$$\boldsymbol{x}_i'(a) = \boldsymbol{x}_i + a\boldsymbol{n}_i, \tag{1}$$

where n_i is the normal at $\boldsymbol{x}_i \in X$ and a is the expansion parameter.

Score. If we apply a dissimilarity metric that is popularly used for calculating the distance between two point clouds (e.g., chamfer distance) as the score, the points in the pocket that are irrelevant to the binding with the inhibitor (far from the inhibitor) inappropriately increase the score. To prevent that, we propose Truncated chamfer distance as score. Truncated chamfer distance with expansion transformation of point cloud X and Y is given by

$$\begin{aligned}&\mathrm{TruncatedCD}(X, Y, a_{\mathrm{est}}, R_{\mathrm{est}}, \boldsymbol{t}_{\mathrm{est}})\\ &\quad=\frac{1}{2}\left(\frac{1}{|X|}\sum_{\boldsymbol{x}\in X}\min\left(\min_{\boldsymbol{y}\in Y}\|R_{\mathrm{est}}\boldsymbol{x}' + \boldsymbol{t}_{\mathrm{est}} - \boldsymbol{y}\|_2,\ d_{\max}\right)\right.\\ &\qquad\left.+\frac{1}{|Y|}\sum_{\boldsymbol{y}\in Y}\min\left(\min_{\boldsymbol{x}\in X}\|R_{\mathrm{est}}\boldsymbol{x}' + \boldsymbol{t}_{\mathrm{est}} - \boldsymbol{y}\|_2,\ d_{\max}\right)\right),\end{aligned} \tag{2}$$

where $\hat{a}_{\mathrm{est}}$ is the estimated expansion parameter, R_{est} is the estimated rotation matrix transformed from quaternion $\boldsymbol{q}$ in $\boldsymbol{z}_{\mathrm{pose}}$, and $\boldsymbol{t}_{\mathrm{est}}$ is the estimated

translation vector from output $\boldsymbol{z}_{\text{pose}}$ by the alignment model. $d_{\max}(= 2.0)$ is the threshold value. In Eq. 2, each point in the expanded inhibitor is rotated and shifted to the estimated pose, and the distance to the closest point in the pocket is accumulated, but distances more than $d_{\max}$ are truncated at that time.

3.3 Ranking

Finally, we obtain the ranking of the candidate inhibitors based on the scores, which indicates how likely the inhibitors bind to the pocket.

3.4 Training of the Proposed Model

This section describes the training method of the proposed model.

Ground Truth for Expansion Parameters. To train the alignment model, we prepare the ground truth (GT) of the expansion parameter a_{gt}. We define it as the expansion that minimizes the distance between the point cloud subset X_{ad} and the pocket Y as follows:

$$a_{\text{gt}} = \arg\min_{a} \frac{1}{|X_{\text{ad}}|} \sum_{\boldsymbol{x} \in X_{\text{ad}}} \min_{\boldsymbol{y} \in Y} \|\boldsymbol{x}' - \boldsymbol{y}\|_2, \tag{3}$$

where X_{ad} is the point cloud subset of inhibitor X whose distances to the closest point $y \in Y$ is less than a threshold, d, and is defined as

$$X_{\text{ad}} = \left\{ \boldsymbol{x} \in X \middle| \min_{\boldsymbol{y} \in Y} \|\boldsymbol{x} - \boldsymbol{y}\|_2 < d \right\}. \tag{4}$$

Loss function. The loss function is written as

$$L = L_{\text{CD}} + L_{\text{rot}} + L_{\text{trans}} + L_{\text{expand}}, \tag{5}$$

where L_{CD} is the chamfer distance between the converted (expanded, rotated, and shifted with the estimated parameters) points in the inhibitor and the pocket point cloud.

$$\begin{aligned} &L_{\text{CD}}(X, Y, a_{\text{est}}, R_{\text{est}}, \boldsymbol{t}_{\text{est}}) \\ &= \frac{1}{2} \left(\frac{1}{|X|} \sum_{\boldsymbol{x} \in X} \min_{\boldsymbol{y} \in Y} \|R_{\text{est}} \boldsymbol{x}' + \boldsymbol{t}_{\text{est}} - \boldsymbol{y}\|_2 + \frac{1}{|Y|} \sum_{\boldsymbol{y} \in Y} \min_{\boldsymbol{x} \in X} \|R_{\text{est}} \boldsymbol{x}' + \boldsymbol{t}_{\text{est}} - \boldsymbol{y}\|_2 \right). \end{aligned} \tag{6}$$

We also give direct supervision to both pose and expansion estimation. L_{rot}, L_{trans} is the error in rotation and translation parameters

$$L_{\text{rot}} = \left| R_{\text{est}} R_{\text{gt}}^{-1} - I \right|_F, \tag{7}$$

$$L_{\text{trans}} = \left\| \boldsymbol{t}_{\text{est}} - \boldsymbol{t}_{\text{gt}} \right\|_2, \tag{8}$$

where R_{gt} is the GT rotation matrix, I is the identity matrix and $|\cdot|_F$ is the Frobenius norm, and $\boldsymbol{t}_{\mathrm{gt}}$ is the GT translation vector.

$L_{\mathrm{expand}}$is the error in the expansion parameter

$$L_{\mathrm{expand}} = \begin{cases} (1+k)|a_{\mathrm{est}} - a_{\mathrm{gt}}| & \text{if } a_{\mathrm{est}} - a_{\mathrm{gt}} > 0, \\ (1-k)|a_{\mathrm{est}} - a_{\mathrm{gt}}| & \text{if } a_{\mathrm{est}} - a_{\mathrm{gt}} \leq 0, \end{cases} \tag{9}$$

where a_{est} is the estimated value of the expansion parameter and a_{gt} is the GT value of the expansion parameter. We apply weights when the estimated expansion parameter is larger than the GT value to prevent the inhibitor point cloud from expanding so much that it overflows the pocket point cloud. k is a constant for weighting and here is set to $k = 0.5$.

4 Experimental Setting

4.1 Datasets

We constructed a dataset of protein-inhibitor pairs for training and validation. First, we listed protein-inhibitor complexes published in ten biochemical journals. Next, we obtained the 3-dimensional structures of the complexes from the Protein Data Bank (PDB) [4] and generated the point clouds of their molecular surfaces. The number of protein-inhibitor pairs obtained was 3,512 for training and 386 for validation. We used the DUD-E dataset [18] for testing, the point clouds of which were obtained in the same way. The number of pairs is 102. The obtained point clouds were sampled to 2,048 points by farthest point sampling [20].

4.2 Training

We used transfer learning from a model pre-trained by ModelNet40 [5]. For pre-training, we generate source and template point clouds from ModelNet40. The template is rotated by three random Euler angles in the range of $[-45°, 45°]$ from the same pose with the source. We train the alignment module by the alignment task between the source and template point clouds. After pre-training, the model was trained by the alignment task between the inhibitor and pocket point clouds, where the initial pose of the inhibitor point cloud was given by rotating it with three random Euler angles $[-45°, 45°]$ from the binding pose.

In the first half of training, we used the loss function without the expansion parameter $L = L_{\mathrm{CD}} + L_{\mathrm{rot}} + L_{\mathrm{trans}}$ to stabilize learning. In the second half of the learning process, the loss function included the error of expansion (Eq.5). When learning the expansion parameter, we apply data augmentation to shrink the inhibitor so that the expansion parameter's GT values become uniformly distributed. The networks were trained for 200 epochs, using a learning rate of 10^{-3}, an exponential decay rate of 0.3 every 50 epochs, and a batch size of 32 in the first and second half of the training. The d_{max} in TruncatedCD was set as the maximum value of the GT expansion parameter, a_{gt}, rounded up to the nearest whole number, $d_{\mathrm{max}} = 2.0$.

5 Alignment and Expansion Evaluation Experiments

In this section, we compared the alignment and expansion estimation accuracy of the proposed and previous methods on the DUD-E dataset.

5.1 Experimental Setting

The pocket point clouds were obtained from the pocket surfaces around the known bound ligands. The binding pose of the pocket point cloud and the inhibitor point cloud was obtained, and the inhibitor point cloud was rotated by three random Euler angles in the range $[-45°, 45°]$ as input. We compared the following methods: registration-based methods using FPFH and RANSAC [21], FPFH and TEASER [28], and feature learning-based methods PointNetLK [2], PCRNet [22], and the proposed method. For the proposed method, we evaluated the impact of the weight sharing in the input embedding module and self-attention layers in terms of accuracy.

The accuracy of alignment was calculated as the rotational error e_{rot} and the translational error e_{trans}

$$e_{\mathrm{rot}} = \theta\left(R_{\mathrm{est}} R_{\mathrm{gt}}^{-1}\right), \tag{10}$$

$$e_{\mathrm{trans}} = \left\|\boldsymbol{t}_{\mathrm{est}} - \boldsymbol{t}_{\mathrm{gt}}\right\|, \tag{11}$$

where $\theta(M)$ is a function that calculates the rotation angle around the rotation axis by considering M as the representation matrix of Rodrigues' rotation formula. R_{est} is the estimated rotation matrix and R_{gt} is the GT rotation matrix.

The accuracy of the expansion was calculated as the expansion error

$$e_{\mathrm{expand}} = |a_{\mathrm{est}} - a_{\mathrm{gt}}|. \tag{12}$$

5.2 Results

Table 1 lists the alignment and expansion estimation accuracy results. We observe that the proposed method had smaller rotational error e_{rot} as well as translational error e_{trans} than the methods compared. Similarly, the expansion error of the proposed method was the smallest without sharing the weights of self-attention layers.

Analysis of Weight Sharing. The proposed method that shared only the weights of input embedding (i.e., did not share the self-sttention weight) had the smallest error. Although the pocket point cloud and the inhibitor candidate point cloud are different objects, their shapes are similar if we focus on the local parts that are in contact with each other. That may be the reason for the performance improvement achieved by sharing the weights of the shallow layers (i.e., the input embedding) which capture local patterns rather than the self-attention weights, which capture global shapes.

Table 1: Accuracy of alignment and expansion estimation

Pose estimate method type	method	Shared weights		e_{rot}	e_{trans}	e_{expand}
		Input embedding	Self-attention			
Registration-Base	FPFH+RANSAC	-	-	120.4	1.504	-
	FPFH+TEASER	-	-	126.9	1.590	-
Learning-Base	PointNetLK	✓	-	32.3	0.721	-
	PCRNet	✓	-	31.5	0.095	-
	Ours	✓	✓	21.9	**0.091**	0.20
		✗	✗	21.6	**0.091**	0.25
		✓	✗	**20.6**	**0.091**	**0.12**

6 SBVS Evaluation Experiments

This section uses the DUD-E dataset to compare the SBVS performance of the proposed and previous methods. In addition, we use ICP to fine-tune the alignment estimated by the proposed method and assess the change in accuracy.

6.1 Experimental Setting

SBVS was performed in the following steps. We docked all candidate inhibitors $X_i (i = 1, 2, \cdots)$ against query pocket Y_j and calculated scores. Then, we obtained retrieval (ranking) results of the inhibitors based on the calculated scores, regarding the pockets as queries. The above retrievals were processed for all query pockets $j = 1, 2, \cdots$. We evaluated the performance of SBVS using the metrics of top-k accuracy and mean average precision at one (mAP@1) defined as

$$\text{mAP@1} = \frac{1}{N_{\text{query}}} \sum_{i=1}^{N_{\text{query}}} \frac{1}{k_i} \tag{13}$$

and processing times. Here, N_{query} is the total number of queries, and k_i is the rank of the correct inhibitor retrieved for the ith query. Processing time is the retrieval time per query. We used six alignment methods (FPFH+RANSAC, FPFH+TEASER, PointNetLK, PCRNet, the propsed method, and the proposed method+ICP) for docking and compared them. For score calculation, we used chamfer distance (CD) and the proposed TruncatedCD. We also compared the SBVS performance with AutoDock Vina, which is based on chemical property optimization.

6.2 Result

Table 2 shows the evaluation results of SBVS accuracy. The proposed model yielded the highest accuracy. Although its processing time was longer than those of PointNetLK and PCRNet; however, its processing time was reduced by about 14/15 compared to AutoDock Vina.

Table 2: Accuracy of SBVS

Method	Score Function	Expansion Estimation	Top-1 Acc.	Top-5 Acc.	Top-10 Acc.	mAP@1	CPU Run Time[s]
AutoDock Vina	-	-	0.28	0.50	0.58	0.39	1,411
FPFH+RANSAC	CD	-	0.01	0.06	0.09	0.06	347
FPFH+TEASER	CD	-	0.01	0.05	0.10	0.06	52
PointNetLK	CD	-	0.03	0.12	0.25	0.10	13
PCRNet	CD	-	0.08	0.024	0.39	0.17	**10**
Ours	CD	✗	0.10	0.25	0.44	0.20	25
		✓	0.33	0.65	0.77	0.48	25
	TruncatedCD	✗	0.11	0.22	0.37	0.19	25
		✓	0.51	0.75	0.78	0.61	25
Ours+ICP	CD	✗	0.10	0.22	0.34	0.18	45
		✓	0.41	0.67	0.76	0.52	43
	TruncatedCD	✗	0.04	0.14	0.25	0.12	44
		✓	**0.57**	**0.75**	**0.81**	**0.65**	44

Fig.5 shows an example of a successful SBVS retrieval using the proposed method that yielded the highest accuracy. The correct pocket-inhibitor correspondence occupied the top-1 rank because of the synergism between estimated expansion transformations and alignment.

Fig.6 shows an example of a SBVS retrieval failure. Fig. 6b and 6c show the ground-truth pose and the output of the proposed model with the correct pocket-inhibitor in Fig.6. We can see that the gap size is not uniform (although the bottom part of the inhibitor is touching the pocket, there are open spaces between the inhibitor and the pocket in other parts in Fig. 6b.). Because the proposed method can deal with only globally uniform expansions, the estimated pose did not fit the pocket, and the score inappropriately increased. Extending the proposed method to support more flexible expansion will be one of our future works.

Ablation Study. As shown in Table 2, the proposed method with ICP, TruncatedCD, and expansion estimation achieved the highest accuracy. We can see that the accuracy of the proposed method with TruncatedCD is higher than that with CD. This is because the truncation removes the score (distance) cal-

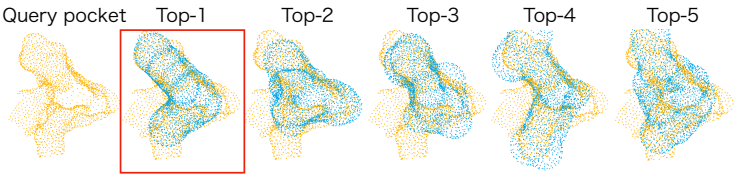
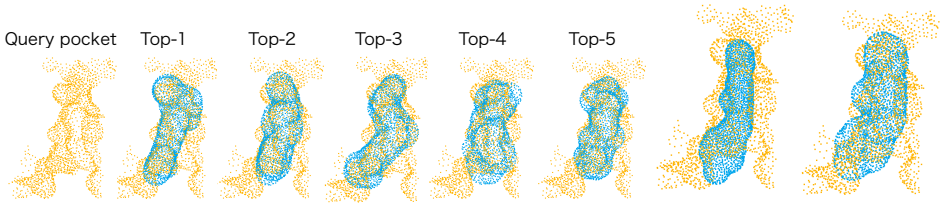


Fig. 5: An example of a successful SBVS retrieval



(a) Ranking result **(b)** GT pose when bounding **(c)** Estimate result (Top-11)

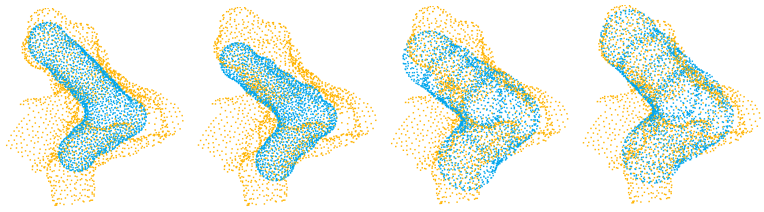
Fig. 6: An example of SBVS retrieval failure

culation between the points that are too far from each other (i.e., irrelevant to the docking) as discussed in Sec. 3.2. In addition, incorporating ICP improved the accuracy of the proposed method because the estimated poses are fine-tuned by ICP and become more accurate.

Fig. 7 shows the intermediate results of the correct inhibitor retrieved in Fig. 5. Fig. 7a, 7b, 7c, and 7d show the ground-truth pose, the alignment estimated by the proposed alignment model, after expansion transformation, and fine-tuning by ICP, respectively.

7 Conclusions

In this paper, we proposed a fast SBVS method based on shape features. The proposed method employs docking simulation based on a learning-based alignment model that simultaneously estimates pose and expansion parameters, and scoring based on Truncated Chamfer Distance with expansion transformation. The proposed method achieved more accurate estimation results compared to previous methods, including chemical property optimization, in less time. Thus, the results suggest that SBVS can be performed quickly and with sufficient accuracy by focusing only on shape features.



(a) GT pose when bounding **(b)** Alignment estimated by the proposed model **(c)** Expansion estimated by proposed model **(d)** Fine-tuned alignment using expansion and ICP

Fig. 7: An example of expansion estimation and alignment fine-tuning

However, newer SBVS methods using graph neural networks and diffusion models have been proposed in recent years. Their performance will need to be compared with that of the proposed method in the future.

References

1. Allen, W.J., Balius, T.E., Mukherjee, S., Brozell, S.R., Moustakas, D.T., Lang, P.T., Case, D.A., Kuntz, I.D., Rizzo, R.C.: DOCK 6: Impact of new features and current docking performance. Journal of computational chemistry **36**(15), 1132–1156 (2015)
2. Aoki, Y., Goforth, H., Srivatsan, R.A., Lucey, S.: PointNetLK: Robust & Efficient Point Cloud Registration Using PointNet. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7163–7172 (2019)
3. Arun, K.S., Huang, T.S., Blostein, S.D.: Least-squares fitting of two 3-d point sets. IEEE Transactions on pattern analysis and machine intelligence (5), 698–700 (1987)
4. Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The Protein Data Bank. Nucleic Acids Research **28**(1), 235–242 (2000)
5. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: ShapeNet: An Information-Rich 3D Model Repository. arXiv preprint arXiv:1512.03012 (2015)
6. Cheeseright, T.J., Mackey, M.D., Melville, J.L., Vinter, J.G.: FieldScreen: Virtual Screening Using Molecular Fields. Application to the DUD Data Set. Journal of Chemical Information and Modeling **48**(11), 2108–2117 (2008)
7. Corso, G., Stärk, H., Jing, B., Barzilay, R., Jaakkola, T.: DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking. arXiv preprint arXiv:2210.01776 (2022)
8. Douguet, D., Payan, F.: Sensaas (sensitive surface as a shape): utilizing open-source algorithms for 3d point cloud alignment of molecules. arXiv preprint arXiv:1908.11267 (2019)
9. Eberhardt, J., Santos-Martins, D., Tillack, A.F., Forli, S.: AutoDock Vina 1.2. 0: New docking methods, expanded force field, and python bindings. Journal of Chemical Information and Modeling **61**(8), 3891–3898 (2021)
10. Eguida, M., Rognan, D.: A Computer Vision Approach to Align and Compare Protein Cavities: Application to Fragment-based Drug Design. Journal of Medicinal Chemistry **63**(13), 7127–7142 (2020)
11. Friesner, R.A., Murphy, R.B., Repasky, M.P., Frye, L.L., Greenwood, J.R., Halgren, T.A., Sanschagrin, P.C., Mainz, D.T.: Extra Precision Glide: Docking and Scoring Incorporating a Model of Hydrophobic Enclosure for Protein − Ligand Complexes. Journal of Medicinal Chemistry **49**(21), 6177–6196 (2006)
12. Gimeno, A., Ojeda-Montes, M.J., Tomás-Hernández, S., Cereto-Massagué, A., Beltrán-Debón, R., Mulero, M., Pujadas, G., Garcia-Vallvé, S.: The Light and Dark Sides of Virtual Screening: What Is There to Know? International Journal of Molecular Sciences **20**(6), 1375 (2019)
13. Goodsell, D.S., Sanner, M.F., Olson, A.J., Forli, S.: The AutoDock suite at 30. Protein Science **30**(1), 31–43 (2021)
14. Guo, M.H., Cai, J.X., Liu, Z.N., Mu, T.J., Martin, R.R., Hu, S.M.: PCT: Point cloud transformer. Computational Visual Media **7**(2), 187–199 (2021)

15. Hawkins, P.C., Skillman, A.G., Nicholls, A.: Comparison of Shape-Matching and Docking as Virtual Screening Tools. Journal of Medicinal Chemistry **50**(1), 74–82 (2007)
16. Maia, E.H.B., Assis, L.C., De Oliveira, T.A., Da Silva, A.M., Taranto, A.G.: Structure-Based Virtual Screening: From Classical to Artificial Intelligence. Frontiers in chemistry **8**, 343 (2020)
17. McNutt, A.T., Francoeur, P., Aggarwal, R., Masuda, T., Meli, R., Ragoza, M., Sunseri, J., Koes, D.R.: GNINA 1.0: molecular docking with deep learning. Journal of cheminformatics **13**(1), 1–20 (2021)
18. Mysinger, M.M., Carchia, M., Irwin, J.J., Shoichet, B.K.: Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. Journal of medicinal chemistry **55**(14), 6582–6594 (2012)
19. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017)
20. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. Advances in neural information processing systems **30** (2017)
21. Rusu, R.B., Blodow, N., Beetz, M.: Fast Point Feature Histograms (FPFH) For 3D Registration. In: 2009 IEEE international conference on robotics and automation. pp. 3212–3217. IEEE (2009)
22. Sarode, V., Li, X., Goforth, H., Aoki, Y., Srivatsan, R.A., Lucey, S., Choset, H.: PCRNet: Point Cloud Registration Network using PointNet Encoding. arXiv preprint arXiv:1908.07906 (2019)
23. Stärk, H., Ganea, O., Pattanaik, L., Barzilay, R., Jaakkola, T.: EquiBind: Geometric Deep Learning for Drug Binding Structure Prediction. In: International Conference on Machine Learning. pp. 20503–20521. PMLR (2022)
24. Su, M., Yang, Q., Du, Y., Feng, G., Liu, Z., Li, Y., Wang, R.: Comparative Assessment of Scoring Functions: The CASF-2016 Update. Journal of Chemical Information and Modeling **59**(2), 895–913 (2019)
25. Trott, O., Olson, A.J.: AutoDock Vina: Improving The Speed and Accuracy of Docking with A New Scoring Function, Efficient Optimization, and Multithreading. Journal of computational chemistry **31**(2), 455–461 (2010)
26. Wang, X., Ramírez-Hinestrosa, S., Dobnikar, J., Frenkel, D.: The Lennard-Jones potential: when (not) to use it. Physical Chemistry Chemical Physics **22**(19), 10624–10633 (2020)
27. Wolber, G., Langer, T.: Ligandscout: 3-d pharmacophores derived from protein-bound ligands and their use as virtual screening filters. Journal of Chemical Information and Modeling **45**(1), 160–169 (2005)
28. Yang, H., Shi, J., Carlone, L.: TEASER: Fast and Certifiable Point Cloud Registration. IEEE Transactions on Robotics **37**(2), 314–333 (2020)
29. Zhou, Q.Y., Park, J., Koltun, V.: Open3D: A modern library for 3D data processing. arXiv:1801.09847 (2018)