HAL open science

Surface-based protein domains retrieval methods from a SHREC2021 challenge

Florent Langenfeld, Tunde Aderinwale, Charles Christoffer, Woong-Hee Shin, Genki Terashi, Xiao Wang, Daisuke Kihara, Halim Benhabiles, Karim Hammoudi, Adnane Cabani, et al.

To cite this version:

Florent Langenfeld, Tunde Aderinwale, Charles Christoffer, Woong-Hee Shin, Genki Terashi, et al.. Surface-based protein domains retrieval methods from a SHREC2021 challenge. Journal of Molecular Graphics and Modelling, 2022, 111, pp.108103. 10.1016/j.jmgm.2021.108103 . hal-03538936

HAL Id: hal-03538936

https://hal.science/hal-03538936v1

Submitted on 8 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Version of Record: https://www.sciencedirect.com/science/article/pii/S1093326321002746
Manuscript_2fa1772b72c9b2c7222acc329ff3cc8e

Journal of Molecular Graphics & Modelling (2021)

Contents lists available at ScienceDirect

Journal of Molecular Graphics & Modelling

journal homepage: www.sciencedirect.com/journal/journal-of-molecular-graphics-and-modelling

Surface-based protein domains retrieval methods from a SHREC2021 challenge

Florent Langenfeld[a,*], Tunde Aderinwale[b], Charles Christoffer[b], Woong-Hee Shin[c], Genki Terashi[d], Xiao Wang[b], Daisuke Kihara[b,d], Halim Benhabiles[e], Karim Hammoudi[f,g], Adnane Cabani[h], Feryal Windal[e], Mahmoud Melkemi[f,g], Ekpo Otu[i], Reyer Zwiggelaar[i], David Hunter[i], Yonghuai Liu[j], Léa Sirugue[a], Huu-Nghia H. Nguyen[k,l,**], Tuan-Duy H. Nguyen[k,l,**], Vinh-Thuyen Nguyen-Truong[k,l,**], Danh Le[k,l,**], Hai-Dang Nguyen[k,l], Minh-Triet Tran[k,l,m], Matthieu Montès[a,*]

[a] *Laboratoire de Génomique, Bio-informatique et Chimie Moléculaire (GBCM), EA 7528, Conservatoire National des Arts-et-Métiers, HESAM Université, 2, rue Conté, Paris 75003, France*
[b] *Department of Computer Science, Purdue University, West Lafayette, IN, 47907, USA*
[c] *Department of Chemical Science Education, Sunchon National University, Suncheon 57922, Republic of Korea*
[d] *Department of Biological Sciences, Purdue University, West Lafayette, IN, 47907, USA*
[e] *Univ. Lille, CNRS, Centrale Lille, Univ. Polytechnique Hauts-de-France, Junia, UMR 8520 - IEMN - Institut d'Electronique de Microélectronique et de Nanotechnologie, F-59000 Lille, France*
[f] *Université de Haute-Alsace, Department of Computer Science, IRIMAS, F-68100 Mulhouse, France*
[g] *Université de Strasbourg, France*
[h] *Normandie University, UNIROUEN, ESIGELEC, IRSEEM, 76000 Rouen, France*
[i] *Department of Computer Science, Aberystwyth University, Aberystwyth, SY23 3FL, UK*
[j] *Department of Computer Science, Edge Hill University, Ormskirk, L39 4QP, UK*
[k] *University of Science, VNU-HCM, Vietnam*
[l] *Vietnam National University, Ho Chi Minh City, Vietnam*
[m] *John von Neumann Institute, VNU-HCM, Vietnam*

ARTICLE INFO

Article history:
Received November 29, 2021

Keywords: SHREC2021, Proteins surface, *2000 MSC:* 92-08,

ABSTRACT

Proteins are essential to nearly all cellular mechanism and the effectors of the cells activities. As such, they often interact through their surface with other proteins or other cellular ligands such as ions or organic molecules. The evolution generates plenty of different proteins, with unique abilities, but also proteins with related functions hence similar 3D surface properties (shape, physico-chemical properties, ...). The protein surfaces are therefore of primary importance for their activity. In the present work, we assess the ability of different methods to detect such similarities based on the geometry of the protein surfaces (described as 3D meshes), using either their shape only, or their shape and the electrostatic potential (a biologically relevant property of proteins surface). Five different groups participated in this contest using the shape-only dataset, and one group extended its pre-existing method to handle the electrostatic potential. Our comparative study reveals both the ability of the methods to detect related proteins and their difficulties to distinguish between highly related proteins. Our study allows also to analyze the putative influence of electrostatic information in addition to the one of protein shapes alone. Finally, the discussion permits to expose the results with respect to ones obtained in the previous contests for the extended method. The source codes of each presented method have been made available online.

© 2021 Elsevier B.V. All rights reserved.

*Corresponding authors
**Equal contributions

e-mail: taderinw@purdue.edu (Tunde Aderinwale),

© 2021 published by Elsevier. This manuscript is made available under the CC BY NC user license
https://creativecommons.org/licenses/by-nc/4.0/

1. Introduction

Proteins are key molecular effectors at the cellular level. Proteins are linear assemblies of amino-acids that fold in specific, energy-driven 3D structures [1, 2] linked to their activity. Identifying similarities within protein structures is therefore of tremendous importance in various fields, from biochemistry to drug design. Numerous methods have been dedicated to structural similarity search of proteins in structural bioinformatics, that mainly rely on the analysis of the 3D point clouds defined by the 3D coordinates of their individual atoms [3, 4, 5, 6, 7]. These methods are mostly based on the conserved core structure of proteins, and therefore, may be inefficient to detect proteins sharing similar surface. The protein surface is a higher-level description of the protein structure that abstracts the underlying protein sequence, structure and fold into a continuous shape with geometric and chemical features that fingerprint its interactions with the other molecules of its environment (solvent, ligands, proteins, nucleic acids, . . .) [8]. Methods able to detect protein surficial similarities are then of major importance.

Only a limited number of methods have been proposed so far:

- Sael *et al.* use 3D Zernike descriptors to detect either global or local similarities between protein's surfaces [9]. This method is able to use surficial physico-chemical properties like the electrostatic potential or the hydrophobicity [10].
- MaSIF (molecular surface interaction fingerprinting) is a geometric deep learning framework that allows to fingerprint biomolecular surfaces [11]. Both geometric and chemical features are extracted and embedded into numerical vectors which is subsequently processed in an application-dependent manner.
- FTIP (Farthest point sampling-enhanced Triangulation-based Iterative-closest-Point) is a global protein surface comparison method that uses the Farthest point sampling method to extract a subset of protein surfaces, and then uses a triangulation-based efficient Iterative-closest-Point algorithm to align these so-called feature-points [12].
- BioZernike [13] adopts a slightly different approach: instead of using the 3D point cloud formed by the atoms coordinates, it uses the density distribution. A 3D Zernike moment normalization procedure is applied to orient the density volumes to be compared, allowing for for fast retrieval of proteins/protein assemblies.

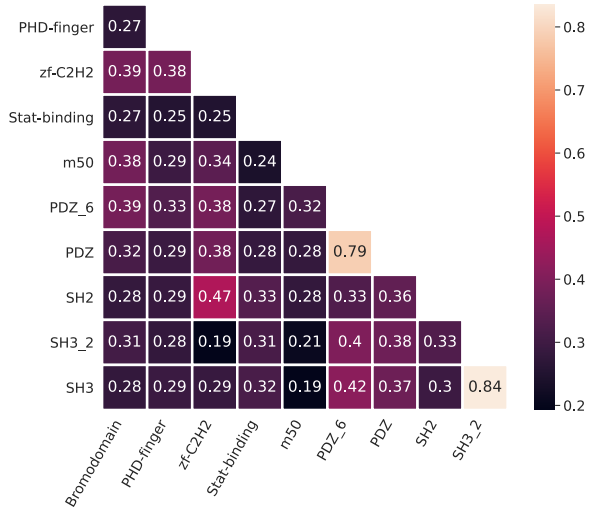


Fig. 1. Structural similarity between the protein structure queries. The TM-score (in the (0, 1] range) measures the topological similarity between two protein structures: the higher the TM-score, the more similar the two structures. Scores below 0.17 correspond to unrelated proteins, while those above 0.5 usually indicate two structures having the same fold [15].

The aim of this track is therefore to assess the performance of currently available methods and to stimulate the development of novel methods. To this end, the dataset encompasses (1) A variety of protein domains, with some of them closely related, to query the dataset. (2) A dataset of experimental structures that contain one or more domains. (3) A few protein shapes corresponding to protein that contains two of the query domains. (4) Two versions of the same dataset, one made of the protein shapes only, and the other with an additional physico-chemical property, the electrostatic potential, encoded along the shapes. We selected this surficial physico-chemical feature as it is the main driving force in many biological recognition processes, such protein-ligand and protein-protein interactions [14]. In the present work, we detail the dataset proposed by the challenge organizers to the participants and how it differs from the previously proposed datasets in Section 2. In Section 3, we describe the 5 methods submitted to the contest. The evaluation metrics are briefly introduced in Section 4, and the performance of the methods is presented in Section 5. Finally, we discuss the outcomes of the different submitted methods in Section 6.

2. The Dataset

2.1. Constitution of the SHREC'21 dataset

The SHREC'21 protein dataset is based on the Pfam 33.1 database [16]. Basically, this database classifies protein sequences into domains and families, that can be grouped into clans whenever they are evolutionarily related. Protein domains of structures from the Protein Data Bank (PDB [17, 18])

christ35@purdue.edu (Charles Christoffer), rainmaker1207@gmail.com (Woong-Hee Shin), gterashi@purdue.edu (Genki Terashi), wang3702@purdue.edu (Xiao Wang), dkihara@purdue.edu (Daisuke Kihara), halim.benhabiles@junia.com (Halim Benhabiles), karim.hammoudi@uha.fr (Karim Hammoudi), adnane.cabani@esigelec.fr (Adnane Cabani), feryal.windal@junia.com (Feryal Windal), mahmoud.melkemi@uha.fr (Mahmoud Melkemi), eko@aber.ac.uk (Ekpo Otu), rrz@aber.ac.uk (Reyer Zwiggelaar), dah56@aber.ac.uk (David Hunter), liuyo@edgehill.ac.uk (Yonghuai Liu), jeremy.sirugue@lecnam.net (Léa Sirugue), nhtduy@apcs.vn (Tuan-Duy H. Nguyen), nhdang@selab.hcmus.edu.vn (Hai-Dang Nguyen), tmtriet@fit.hcmus.edu.vn (Minh-Triet Tran), florent.langenfeld@lecnam.net (Florent Langenfeld), matthieu.montes@cnam.fr (Matthieu Montès)

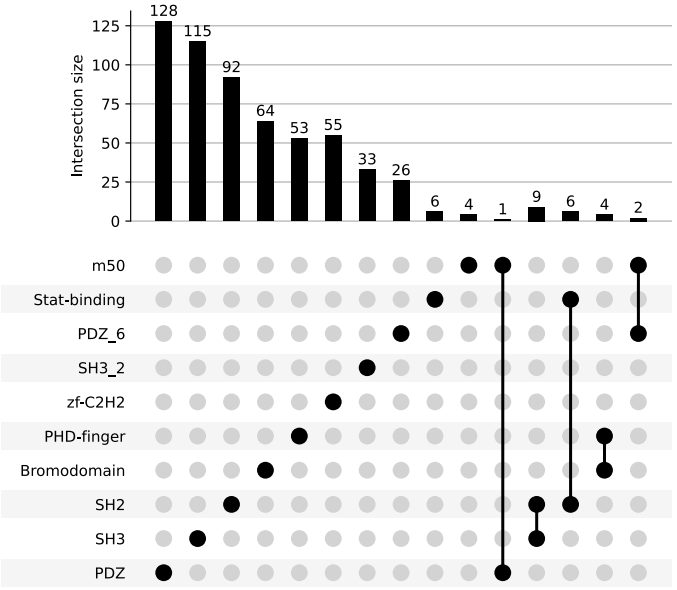


Fig. 2. The Upset Plot of ten selected Pfam domains in SHREC2021 challenge datasets. The dataset is composed of 554 individual shapes, of which 22 bears two of the domains of the dataset.

can therefore be attributed to a Pfam domain and, possibly, a clan. To build up the track dataset, we relied on this notion of domain, and manually selected 10 Pfam domains: the SH2 domain (PfamID PF00017), the SH3 domain (PfamID PF00018), the variant SH3 domain (SH3_2, PfamID PF07653), the PDZ domain (PfamID PF00595), the PDZ_6 domain (PfamID PF17820), the peptidase family M50 (m50, PF02163), the bromodomain family (PF00439), the DNA-binding domain of the STAT protein (STAT-binding, PF02864), the PHD-finger domain (PfamID PF000628), and the C2H2 Zinc-finger domain (zf-C2H2, PfamID PF00096).

For each selected domain, all corresponding structures from the PDB were listed, and the best resolution structures were retrieved to serve as a query for the track. When applicable, the NMR (Nuclear Magnetic Resonance) structures were assigned an arbitrary resolution of 2.25 Å [19], while structures with no resolution were discarded. The residues corresponding to the Pfam domains were then extracted from the selected structures when necessary, so that the selected domains were left alone. For example, only the DNA-binding domain of the STAT (Signal Transducer and Activator of Transcription) protein was kept as a query, its others domains being discarded.

The remaining structures were filtered according to their Uniprot [20] identifier, and duplicates were discarded to (1) Ensure a diversity of sequence structures among the dataset. (2) Limit the dataset size to a tractable size given the track timeline. Finally, only the best resolution structures for each Uniprot entry were kept. When NMR structures were selected, only the first model was considered. Unlike the query structures, we kept the other domains present in these structures that eventually constitute the dataset. Therefore, many dataset structures display several domains, at least one of which is one of the query domains.

For all structures (queries and dataset structures), we removed all hetero-atoms, and unwanted chains. The resulting PDB structures were then protonated using pdb2pqr [21], using propka [22, 23] to compute the pKa values of the ionizable groups at pH=7.2. The solvent-excluded surface of all protonated structures were computed using the default parameters of EDTSurf [24, 25] except that inner cavities were discarded. We then computed the electrostatics using APBS suite [26], and used the *multivalue* to compute the electrostatic potential at the mesh vertices locations. Two datasets were then assembled, one with only the protein surface shapes (shape-only dataset) and one combining the protein surface shapes and electrostatics values (shape+electrostatic dataset). Similarly, two sets of query surfaces were produced (shape-only and shape+electrostatic). Each dataset (shape-only and shape+electrostatic) includes 554 molecular surfaces which were made available to the track participants, along with the 2 sets of 10 queries, on the track web–page (`http://shrec2021.drugdesign.fr`).

Regarding the dataset, it is important to note that SH3 and SH3_2 domains were annotated as similar according to HHsearch (a tool commonly used to detect homologous proteins [27]), as well as the PDZ, PDZ_6 and m50 domains. We present in Figure 1 the TM-scores matrix for all queries of the dataset. A TM-score above 0.5 indicates that the two structures are likely to share the same topology, while unrelated structures are usually associated to TM-scores below 0.17 [28]. SH3 and SH3_2 query structures show a TM-score of 0.84, while, PDZ and PDZ_6 query structures show a TM-score of 0.79. Surprisingly, the m50 query structure only has a TM-score of 0.28 and 0.32 with PDZ and PDZ_6 structures, respectively. A visual inspection of these structures confirmed that the peptidase M50 is topologically different from both the PDZ and PDZ_6 domains: while the peptidase M50 is mainly α-helical, PDZ and PDZ_6 are mixed $\alpha-\beta$ proteins. Overall, most query structures present an intermediate topological similarity with all other queries, as evidenced by the fact that all TM-scores range from 0.19 to 0.47, except for the aforementioned pairs of classes (SH3 / SH3_2 and PDZ / PDZ_6).

Finally, as the dataset encompasses multi-domain proteins, 22 dataset proteins display two of the query domains. Namely, 9 proteins of the dataset encompass both a SH2 and a SH3 domains, 6 proteins encompass both a SH2 and a Stat-binding domains, 4 proteins encompass both a bromodomain and a PHD-finger domain, 2 proteins encompass a PDZ_6 and a peptidase family M50 domains, and one structure encompass a PDZ and a peptidase m50 domains. The final structure of the dataset and the size of each class is summarized in Figure 2.

2.2. *Comparison to previous SHREC datasets on proteins*

Compared to previous SHREC datasets dealing with protein structures or surfaces [29, 30, 31, 32, 33, 34], the SHREC'21 protein track dataset is characterized by three main aspects : (1) The presence of two datasets representing the same set of proteins, one shape-only and one shape+electrostatic dataset. (2) The close evolutionary relationship of some of the query domains, further characterized by a similar topology (Figure 1). (3) The intermediate similarity of the domains topologies (Figure 1). And (4) The use of individual domains to query a dataset of single- as well as multi-domain proteins shapes.

The main novelty of the SHREC'21 track is arguably the availability of protein surfaces with electrostatic values, which has been shown to improve the retrieval performance of protein surfaces [11, 35]. This additional feature might therefore allow to better distinguish structurally related proteins based on their surficial properties and improve the methods' performance.

2.3. Challenge proposed to the participants

SHREC, or 3D Shape Retrieval Challenges, are challenges primarily organized in order to evaluate the effectiveness of 3D-shape retrieval algorithms. A group organizes a challenge by building up a dataset, then proposes the challenge publicly to the community, and finally gathers, analyses and verifies the results. The theme of the challenge may vary from one to another, but all challenges take place in a limited time, which ranges from 1 to 1 ½ months.

In our contest, the participants were asked, given each of the 10 query surfaces, to retrieve the molecular surfaces of proteins from the dataset that encompass the same domain as the query. Each query-to-dataset-surface distance was expected to be expressed as a dissimilarity score. The results were therefore 10×554 matrices of dissimilarity scores. Each participant was allowed to submit one dissimilarity matrix for each dataset: one matrix for the shape-only dataset, and one matrix for the shape+electrostatic dataset.

3. Participants and methods

Among the seven groups that initially registered to this track, only 5 were able to produce the results in time and returned a shape-only dissimilarity matrix. Only one method (3DZD, see 3.1) returned a dissimilarity matrix for the shape+electrostatic dataset. The other groups were not able to produce a satisfying training dataset or willing to develop their algorithm to handle the electrostatics values. In the following subsections, each group describes their new, respective methods.

3.1. Network trained with encoded 3DZD (3DZD) by Tunde Aderinwale, Charles Christoffer, Woong-Hee Shin, Genki Terashi, Xiao Wang & Daisuke Kihara

Our group submitted two (shape-only and shape+electrostatic) dissimilarity matrices of the target proteins to the 10 query proteins provided by the organizers. These methods are based on the 3D Zernike Descriptor (3DZD). 3DZD is the rotation-invariant shape descriptor derived from the coefficients of 3D Zernike-Canterakis polynomials [36].

3.1.1. Summary of the 3DZD method

Similar to SHREC'20 [34], our group trained two types of neural network to output a score that measures the dissimilarity between a pair of protein shapes. Briefly, the first framework (the Extractor model) is structured into multiple layers: an encoder layer with 3 hidden units of size 250, 200 and 150, a feature comparator layer which computes the Euclidean distance, cosine distance, element-wise absolute difference and product, and a fully connected layer with 2 hidden units of size 100 and 50. There are multiple hidden units in each layer, and our group uses the ReLu activation function in all except the output of the fully connected layer where the sigmoid activation function is used to output the probability that the two proteins belong to the same protein– or species–level in the SCOPe dataset classification [37, 38]. The second framework (the end–to–end model) is similar to the first except the feature comparator layer is removed and the output of the encoder is directly connected to the fully connected layer.

The network is trained on the latest SCOPe dataset of 259,385 protein structures. 2,500 protein structures were set aside for network validation. Proteins in Class I (Artifacts) were removed. Each of the two network frameworks is trained with two datasets. The first dataset is 3DZDs of the surface shape of proteins and the second one is feature vectors that concatenate 3DZD of shape and 3DZD of the electrostatic properties.

Our group examined the performance of the networks on the validation dataset to determine which models to use. For the shape-only dataset, we submitted predictions generated by the Extractor model. For the shape+electrostatic dataset, we submitted the average predictions between the Extractor model and the end–to–end model.

For each protein in the provided dataset, our group performs a pre–processing step as follows: (1) The PLY mesh data file is converted to a volumetric skin representation (Situs file) where points within 1.7 grid intervals are assigned with values that are interpolated from the mesh [9]. For the electrostatic features, the interpolated values are the potentials at the mesh vertices. For the shape feature, a constant of 1 is assigned to grids that overlap with the surface. (2) The resulting Situs file is fed into the EM-Surfer pipeline [39] to compute 3DZD.

3.1.2. Runtimes and computational cost

It takes approximately 12–13 min to pre-process each file. Generating 3DZD took ~8.00 s on average for each protein on an Intel® Xeon® CPU E5-2630 0 @ 2.30GHz. The training of each models took 12 h. Dissimilarity prediction between two proteins using the trained model took ~0.22 s on average on a Nvidia® Titan X GPU. The averaging of the two matrices was almost instant and negligible. The code is available at the following url: `https://github.com/kiharalab/shrec_2021_shape_retrieval`.

3.2. ProteinNet: Deep learning based protein characterization from 3D point clouds (ProteinNet) by Halim Benhabiles, Karim Hammoudi, Adnane Cabani, Feryal Windal & Mahmoud Melkemi

Our group proposes a deep learning approach to calculate a protein descriptor from its 3D point cloud. To this end, we developed a variant of PointNet [40] which is a point cloud deep architecture dedicated for 3D classification and segmentation. We adapted this architecture in order to learn an affine transformation matrix that allows to align the coordinates of the input 3D protein point cloud into a canonical representation. The new representation maintains interesting properties demonstrated in [40], including invariance to rigid geometric transformations as well as point order permutations. The ProteinNet deep architecture is illustrated in Figure 3. More specifically, the architecture is based on a PT-Net module (Protein

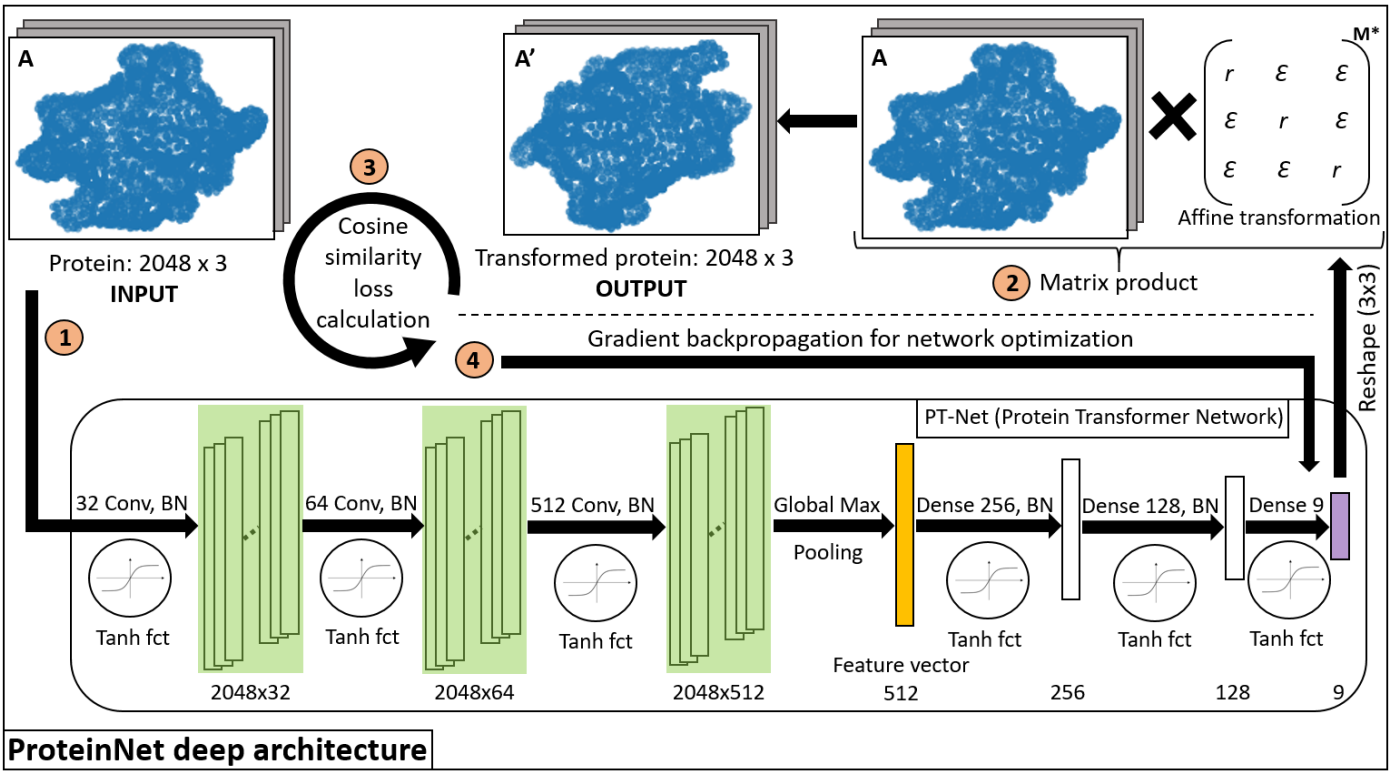


Fig. 3. ProteinNet deep architecture for protein point cloud transformation into canonical representation. Step (1): affine transformation matrix estimation. Step (2): protein point cloud transformation using the estimated affine matrix. Step (3): similarity calculation between the original protein point cloud (the input) and its transformed point cloud. Step (4): cosine similarity loss calculation between the original input protein point cloud and its transformation; and back–propagation over the network to optimize the estimation of the affine transformation matrix.

Transformer Network) which is inspired from the T-Net (Transformer Network) module of the original PointNet architecture. The PT-Net module is trained to predict an affine transformation matrix M that is constrained to be close to an orthogonal matrix, namely $|(M.M^t) - I| = 0$ (step 1 in Figure 3). The matrix M is used to transform the input protein into its canonical representation (step 2 in Figure 3). A cosine similarity loss between the original protein and the transformed one is then calculated (step 3 in Figure 3) in order to back-propagate the error over the network (step 4 in Figure 3) and optimize the matrix M.

3.2.1. *PT-Net module*

The module is composed of a sequence of 3 convolution blocks (32, 64 and 512 layers) followed by a global max pooling layer and 3 successive dense layers (256, 128 and 9). As shown in Figure 3, each convolution block as well as the dense layers (except the last one) undergo a batch normalization and a tangent hyperbolic activation function. The last dense layer of 9 units is reshaped to output the (3×3) M matrix (see details in [40]).

3.2.2. *Data preparation and architecture training*

All the proteins of the dataset of the track have been sampled to 2,048 points using a Poisson disk sampling technique [41] and then normalized into a zero-center unit sphere based on their respective minimum bounding spheres [42]. The architecture has then been trained using a batch size of 16 on 80% of the dataset over 150 epochs and validated on the remaining 20% of the data. The training data were augmented on-the-fly (during the training process) by adding some geometric noise (*e.g.* random displacement of point coordinates in a limited interval).

3.2.3. *Protein feature extractor*

The trained ProteinNet model has then been exploited to calculate a protein feature descriptor, for each input protein, by extracting its intermediate Global Max Pooling hidden layer. This descriptor corresponds to a 1-dimension vector of 512 values.

3.2.4. *Dissimilarity matrix computation*

The dissimilarity matrix between the ten protein shape queries and the set of 554 protein shapes has been calculated using Euclidean distance between their respective 512 feature vectors.

3.2.5. *Runtimes and computational cost*

This framework has been developed in Python 3.7.6 using different libraries, namely Open3D 0.8.0.0, and Keras 2.2.4-tf on a TensorFlow-GPU 2.1.0 backend. The experiments have been conducted on an Intel Xeon® Gold® 5118 CPU@2.30 GHz with 128 GB of memory and NVIDIA® Tesla® T4 GPU with 16 GB of memory. The running times in sof each stage performed on CPU are reported in Table 1 for one protein. Table 2 shows the training times of the ProteinNet model trained on GPU. The code is available at the following url: `https://github.com/Benhabiles-JUNIA/ProteinNet`.

Table 1. Running times in s using CPU for each stage of the ProteinNet framework obtained for one protein.

Point cloud maximum and minimum sizes of two proteins	Point cloud sampling (2,048) and normalization	Feature descriptor (512) calculation of one protein	Distances from one protein to all protein dataset (554 proteins)
582,496 points	1.14	0.005	0.004
37,658 points	0.9		

Table 2. Running times in s using CPU for each stage of the ProteinNet framework obtained for one protein.

Deep learning model	ProteinNet
Training data size	499
Epochs	150
Training time (s)	155

3.3. Fisher Kernel agglomerated local Augmented Point Pair Feature Descriptors, trained with Gaussian Mixture Model (APPFD-FK-GMM) by Ekpo Otu, Reyer Zwiggelaar, David Hunter & Yonghuai Liu

Our group presents a novel APPFD-FK-GMM 3D shape retrieval method (see Figure 4) based on Fisher Kernel (FK) and Gaussian Mixture Model (GMM) agglomeration of the Augmented Point-pair Feature Descriptor (APPFD) [43]: a 3D key point shape descriptor that robustly captures the physical geometric characteristics of 3D surface regions. Previous APPFD binning technique involves bucketting each of the 6-dimensional features of the APPFD into a multi-dimensional histogram with at least 7 bins in each feature-dimension, resulting to approximately $7^6 = 117649$-dimensional final feature-vector (APPFD), which is very high-dimensional final descriptor.

In this work, we contribute a simpler approach, where each of the 6-dimensional feature is binned into a 1-dimensional histogram with 35 bins for each feature-dimension to produce a 210-dimensional local descriptor (APPFD) for every key point or local surface patch (LSP). Finally, the locally computed APPFDs are agglomerated into a compact code called the Fisher Vector (FV) with 4210 dimensions, which is L_2 and power-normalized, and represents a single protein model, using the FK and GMM [44] framework.

The goal of the APPFD-FK-GMM method/contribution is to provide a straight-forward, efficient, robust, and compact representation, describing the geometry of 3D protein surfaces, with a knowledge-based (*i.e.* non-learning) approach. While a single protein surface in this challenge contains an average of $120,000$ vertices and $200,000$ triangular faces, our implementation address this very high data-structure by reducing 3D protein surface representation to $3,500$ points sample.

3.3.1. Summary of The APPFD-FK-GMM Method

Our method involves two main stages: (1) Computing local APPFDs for selected key points on 3D protein surface. (2) Key points APPFDs aggregation with FK and GMM described below. Figure 4 shows the processing pipeline of the APPFD-FK-GMM algorithm with complete implementation details provided in [44]. The reader is referred to [45], for further details regarding this method.

Stage (1) - Computing Local APPFDs. Following key points (p_{k_i}) determination for each 3D protein surface, represented as point cloud (P), the 4-dimensional feature, $f_1 = (\alpha, \beta, \gamma, \delta)$ in [46] is augmented with a locally-extracted 2-dimensional feature: $f_2(p_i, p_j) = (\phi, \theta)$ for every possible combination of point pair, p_i, p_j (*without* their estimated normals, n_i, n_j) in the local surface patch (LSP), $\{P_i, i = 1 : K\}$ around each key point $\{p_{k_i}, i = 1 : K\}$ in P_s, where K is the number of key points. The extraction of f_2 (see Figure 5) is because f_1 is not robust enough to capture the entire geometric details of the underlying surface, whereas, the PPF approach opens up possibilities for additional feature space.

The angular projections: θ and ϕ in Figure 5 are derived by taking the scalar products of $(\overrightarrow{S} \cdot \overrightarrow{V_1})$ for $\angle_1$, and $(\overrightarrow{S} \cdot \overrightarrow{V_2})$ for $\angle_2$ about a point p_i in a given LSP. Mathematically, scalar products defined in this manner are homogeneous (*i.e.* invariant) under scaling and rotation. Therefore, f_2 is considered rotation and scale invariant for 3D shapes under rigid and non-rigid transformations [34].

Finally, a 6-dimensional $f_3 = (f_2 + f_1)$ are locally obtained thus: $f_3(p_i, p_j) = (f_2(p_i, p_j), f_1(p_i, p_j)) = (\phi, \theta, \alpha, \beta, \gamma, \delta)$, and binned into a 1-dimensional histogram with bins = 35 in each feature-dimension, normalized and concatenated to give 35×6 = 210-dimensional single local APPFDs per LSP.

Stage (2) - Key points APPFDs Aggregation with FK and GMM. Here, the final descriptor (*i.e. fisher-vector, FV*) computation approach involves an initial step of training a GMM, given aggregated key points local APPFDs for all database 3D objects, then FK is applied on the trained model and a single protein's local APPFDs to derive a global signature (APPFD-FK-GMM) for the protein surface (see Figure 4).

3.3.2. Runtimes and computational cost

Our group submitted a dissimilarity matrix $D = [10 \times 554]$, where the entry $D = [i, j]$ corresponds to the L_2 distance from i^{th} FV in the *query* set to the jth FV in the *collection* set.

While implementing the APPFD-FK-GMM for this task, K is specified by the parameter, $vs = 0.20$, which is the voxel size for point cloud down-sampling, while the radius parameter, $r = 0.50$ specifies the size of P. Regarding point cloud size, $P = 3,500$ points are sampled.

In conclusion, we present a pure Python 3.60 implementation code that computes the APPFD-FK-GMM method. All

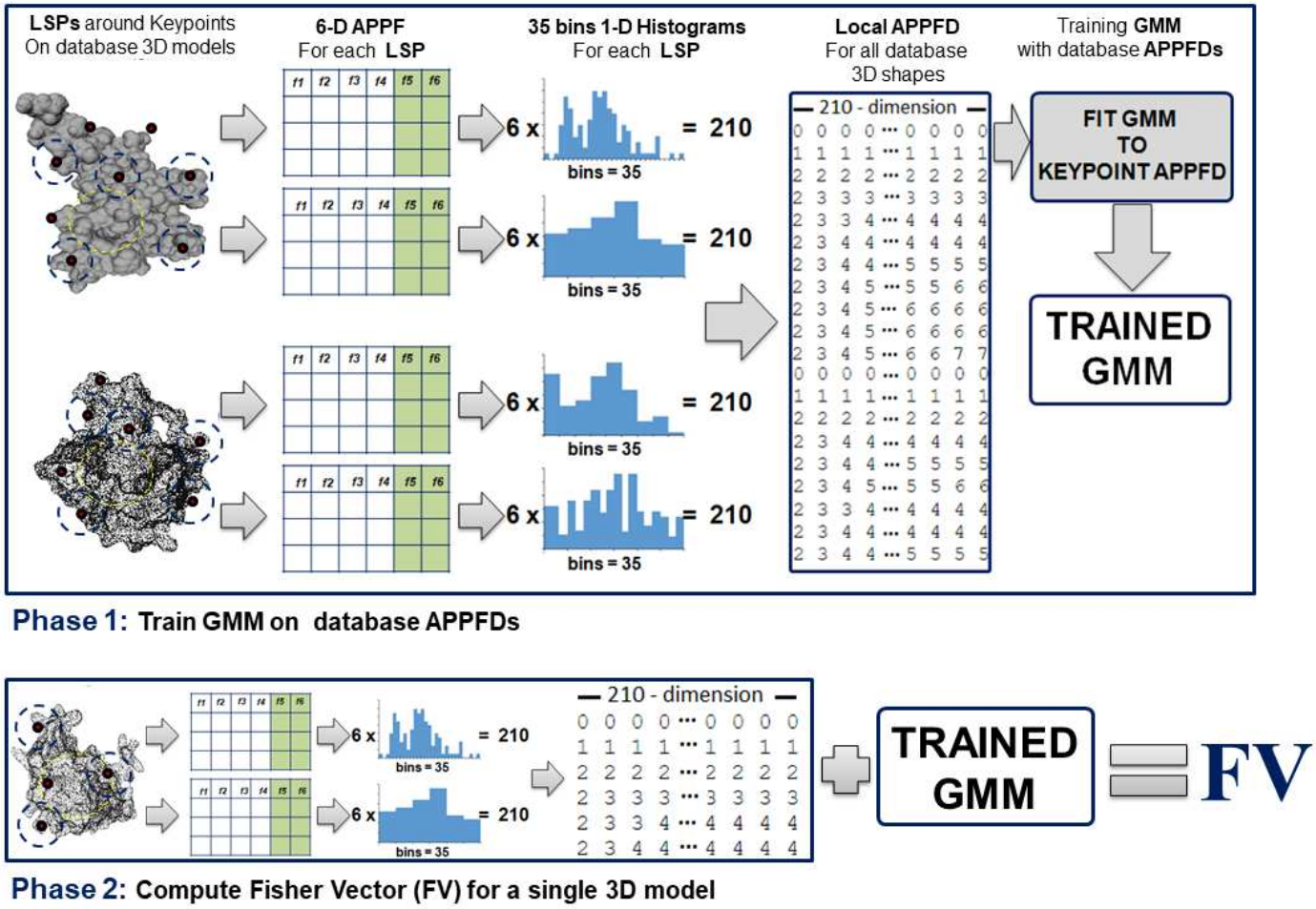


Fig. 4. APPFD-FK-GMM processing pipeline involving Phase 1 (fitting a GMM to all the keypoints or LSPs descriptor, *i.e.* local APPFDs from each 3D protein surface and for all database protein surfaces) and Phase 2 (computing a single compact descriptor called fisher-vector (FV) for each 3D protein by aggregating all its keypoints or local APPFDs using the fisher kernel (FK) framework and the trained GMM in Phase 1. Within each LSP around a keypoint, six different geometric features are first extracted, and each feature-dimension is binned into a 1D histogram with 35 bins, where all histograms are combined to form a 210-dimensional descriptor, *i.e.* local APPFD for each LSP. All such LSP descriptors from each 3D protein are compacted into a 4210-dimensional FV for that protein model, as in Phase 2.

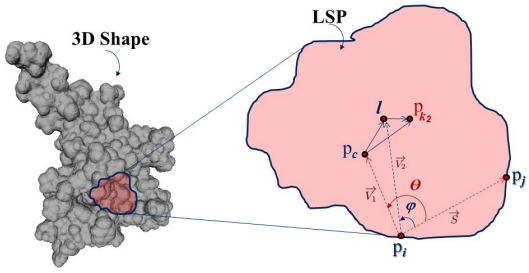


Fig. 5. Local Surface Patch (LSP), P_i with pairwise points (p_i, p_j) as part of a surflet-pair relation for (p_i, n_i) and (p_j, n_j), with p_i being the origin. θ and ϕ are the angles of vectors projection about the origin, p_i. θ is the projection angle from vector $\langle p_i - p_j \rangle$ to vector $\langle p_i - p_c \rangle$ while ϕ is the projection angle from vector $\langle p_i - p_j \rangle$ to vector $\langle p_i - l \rangle$. The LSP centre is given by p_c, keypoint is given as p_{k_i} where $i = 2$. Finally, l is the vector position of $p_{k_i} - p_c$. [34]

experiments were carried out under Windows® 7 desktop PC with Intel® Core® i7-4790 CPU @ 3.60GHz, 32GB RAM. It takes an average of 30 s to compute the APPFD-FK-GMM. The implementation code is available at the following url: `https://tinyurl.com/shrec21`.

3.4. Projected Wave Kernel Signature Maps (PWKSM) by Léa Sirugue & Matthieu Montès

This method is based on the 2D projection of the surface and the Wave Kernel Signature (WKS) descriptor. Wave Kernel Signature [47] is an isometric invariant descriptor that has been extensively improved and used in the field of computer vision [48, 49, 50, 51]. We have combined WKS with a 2D projection on a unit sphere [52]. Lowering one dimension of the space allows us to have a fast and dense comparison of the surface while having a smaller storage size for files.

Descriptor calculation. In a first step, the WKS descriptor is computed on the surface of the 3D object for each point of the

mesh. The surface is flattened on the unit sphere using a conformal transformation [52]. Then, the 2D spherical coordinates of the unit sphere are converted into 2D cartesian coordinates on the plane [53]. A maps of size $(\theta_{max} - \theta_{min})/\delta, (\phi_{max} - \phi_{min})/\delta$ is created. θ_{max} and θ_{min} are the maximum and minimum values of θ on the unit sphere and the same with ϕ, each representing an angle coordinate. δ is a coefficient to adapt for resolution. This type of projection is similar to topographic maps, that is why our group called this descriptor Projected Wave Kernel Signature Maps (PWKSM). An interpolation in the space of discrete integers is done to densify the maps. To reduce impact of deformation at the poles when converting to 2D cartesian coordinates, we computed 7 different maps with different pole orientations.

Descriptor comparison. A dense comparison is made using GPGPU sum reduction technique [54] [55] [56]. Each point's WKS of a PWKSM is compared to all points' WKS of another PWKSM. The Earth Mover's distance L is used to compare the WKS descriptor of each point. Then, the smallest distance between a point of a first PWKSM T and all points of a second PWKSM V is selected. The sum of all the smallest distances for each point of the first PWKSM are summed to create the score S_T. The same is done for computing S_V.

$$S_T(T, V) = \sum_{k_T=1}^{N_T} \min_{k_V} L(T(k_T), V(k_V)) \tag{1}$$

The final score is the average of S_T and S_V defined as follows:

$$S = \frac{S_T + S_V}{2} \tag{2}$$

3.4.1. Runtimes and computational cost

All the calculations were made on a computer based on a 64-bit OS with an Intel® Xeon® CPU @ 2.30GHz, a Nvidia® Quadro® k4200 GPU with 4GB and 32GB of RAM.
Computing the WKS took on average 9 min and 31 s. It required on average 44 s to compute one PWKSM. The comparison of two surfaces (*i.e.* 7 versus 7 PWKSM) takes on average 23 s. The code is available at the following url: `https://gitlab.cnam.fr/gitlab/siruguej/PWKSM`.

3.5. Graph-based learning methods for Surface-based protein domains retrieval (DGCNN) by Huu-Nghia H. Nguyen, Tuan-Duy H. Nguyen, Vinh-Thuyen Nguyen-Truong, Danh Le, Hai-Dang Nguyen & Minh-Triet Tran

In this deep learning method, our group exploits the availability of protein class labels from [35] to optimize the representation of protein surfaces without any additional properties. Particularly, we designed a message-passing graph convolutional neural network (MPGCNN) with the Edge Convolution (EdgeConv) paradigm [57] for the protein classification objective. Then, the latent representation of protein surfaces from this neural network is used for the retrieval task in this track.

3.5.1. Data pre–processing

For the meshes in each 3D model of a protein surface, we first sample 512 points on the surfaces of the meshes based on the area of the meshes. Then, to re-assign the topological structures for sampled points, we connect each nodes with their k-Nearest Neighbors based on their original coordinates ($k = 16$).

3.5.2. Edge Convolution

In this geometry-only setting, the initial node features is the coordinates of sampled points. Each protein surface is represented by a k-Nearest Neighbors graph generated in the pre–processing step with 512 vertices (nodes).

The module that performs the graph message-passing function is the EdgeConv layer [57]. In the EdgeConv layer, the information of a vertex i after layer l is calculated as follows:

$$x_i^{l+1} = max_{j \in N}\, h(x_i^l, x_j^l) \tag{3}$$

where N is the neighboring vertices of vertex i with

$$h(x_i^l, x_j^l) = ReLU(MLP(x_i^l \oplus x_j^l)) \tag{4}$$

where ReLU is Rectified Linear Unit (in this implementation, we used LeakyReLU—a variant of ReLU), MLP is a standard multilayer perceptron (MLP), and $\oplus$ is the concatenation operator.

In this implementation, our group uses a dynamic variant of EdgeConv instead of the standard EdgeConv described above. At each Dynamic EdgeConv layer, each vertex's k–Nearest Neighbors is re-calculated in the feature space produced by the previous layer, before applying the standard EdgeConv operation. After the graph has been recomputed, standard EdgeConv operation is performed.

After the pre–processing phase, the vertex features first go through 4 layers of Dynamic EdgeConv. The dimensions of output features for each vertex after these first-4 layers are $64, 64, 128$, and 256, respectively. Then, the outputs of these 4 layers are concatenated to become a 512-dimensional vector for each vertex. This 512-dimension vector is then fed through another Dynamic EdgeConv layer, creating the output vector v with 512 dimensions. The feature vector v is pooled using the concatenation of the outputs of a *max-pooling* and a *mean pooling* layer to generate the first graph-level feature vector. This vector is passed through two MLP blocks with BatchNorm, Leaky-ReLU, and Dropout layers. Finally, the vector is passed through a Fully-Connected layer for classification.

The latent representation of the graph is extracted as vectors by removing the last Fully-Connected layer from the network. The retrieval task is then performed by exploiting the $L2$-distances between these vectors.

3.5.3. Runtimes and computational cost

This method is implemented in Python 3.8 [58], using Pytorch [59] and Pytorch Geometric [60] libraries. The experiments were carried out a machine with an Intel® Core® i7-8700K 6-core CPU Processor @3.70 GHz with 32 GB of RAM and an NVIDIA® TITAN V with 12 GB of VRAM. The training and test set's embedding extraction uses both the CPU and

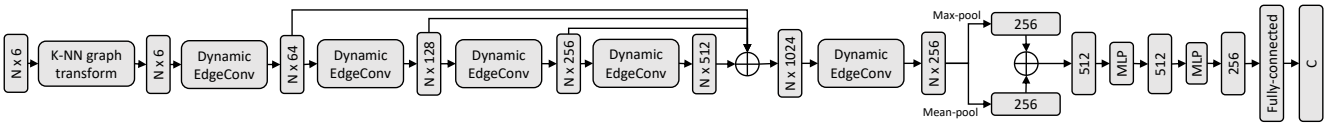


Fig. 6. Dynamic Edge Convolutional Neural Network

the GPU, while computation of distance matrix only uses the CPU. The detailed time report is represented in Table 3. The code is available at the following url: `https://github.com/huunghia160799/SHREC-protein-domains`.

Table 3. Time report of each step of the DGCNN method.

Training	Test Set Extraction	Matrix Computation	Total
≈ 1100 min	≈ 7 min	≈ 3 min	1173 min

4. Evaluation metrics

We use common evaluation metrics to assess the performance of the proposed methods, most of which are used in other SHREC tracks [61] or similar works evaluating the performance in retrieval [62]. For each method, we compute the overall metrics (*i.e.* the metrics averaged over all queries) and the individual metrics (*i.e.* the metrics for each query) to provide a better understanding of the performance of each method for each query. Two composite classes are also presented: the SH3-like and the PDZ-like, which correspond to the grouping of the SH3 and SH3_2 classes, and of the PDZ and PDZ_6 classes, respectively. To set the dissimilarity values for these composite classes, for each entry of the dataset, we kept the minimal dissimilarity value from the SH3 / SH3_2 queries and from the PDZ / PDZ_6 queries.

4.1. Nearest neighbor, First tier and Second Tier

These metrics measures the ratio of relevant objects among the k retrieved objects, and ranges in the interval $[0, 1]$. For the nearest neighbor (NN), only the first retrieved object is considered ($k = 1$), while the top C objects are considered for the first tier (FT), and the top $2 \times C$ objects for the second tier (ST). Here, C represents the cardinal of the class under investigation, *i.e.* the size of the class to which the query belongs. Higher values of nearest neighbor, first tier and second tier indicate better performance.

4.2. Precision-Recall curves

The Precision (P) represents the fraction of relevant object retrieved compared to the top k retrieved objects: $P = (relevant \bigcap retrieved)/retrieved$. Therefore, precision can be evaluated at different intervals. The Recall (R) represents the fraction of relevant objects retrieved compared to the size C of the class of the query: $P = (relevant \bigcap retrieved)/relevant$. Both metrics range from 0 to 1. Precision-Recall curve plots the precision values at given recall values, which produces, in an ideal case, an horizontal line at $P = 1$ that spans the entire range of recall values.

4.3. Confusion matrix

A confusion matrix (CM) is a square matrix, whose columns represents the different classes of the dataset and rows the class of the query. For each row, each element $CM(i, j)$ gives the number of objects from class i retrieved using the query j, considering the top $k = C$ retrieved objects, C being the size of the class corresponding to query j. The elements $CM(i, i)$ in the diagonal of the confusion matrix indicates the objects classified correctly, while the off-diagonal indicates mis–classified elements. To ease the comparison between classes of different sizes, the numbers were normalized over the class sizes C of the queries. Consequently, the sum $\sum_j CM(i, j)$ of all elements of each row equals 1.

4.4. Reciprocal Rank and Mean Reciprocal Rank

The Reciprocal Rank (RR) measures the performance to find the first relevant item. For a given query, it equals to the inverse of the rank r of the first relevant item found: $RR = 1/r$. The Reciprocal Rank ranges from 0 (no relevant object found) to 1 (the first retrieved object is relevant). The Mean Reciprocal Rank (MRR) is the Reciprocal Rank averaged over all queries. This metric is useful as: (1) It is considered order-aware, contrary to the previous metrics. (2) Typical use cases only consider the few first retrieved items; therefore, the higher the reciprocal rank, the better the performance.

5. Results

Among the participants of the track, all teams returned a dissimilarity matrix for the shape-only dataset, and only one method (3DZD) was adapted to handle the shape+electrostatics dataset.

5.1. Shape-only challenge

The results for the shape-only dataset are presented in Table 4 and in Figures 8 and 7. Table 4 summarizes the performances of all submitted matrices for the shape-only dataset. For each metric (Nearest Neighbor, First Tier, Second Tier and Mean Reciprocal Rank), the highest value is indicated in bold. Given the dataset structure and the selected query domains, the best

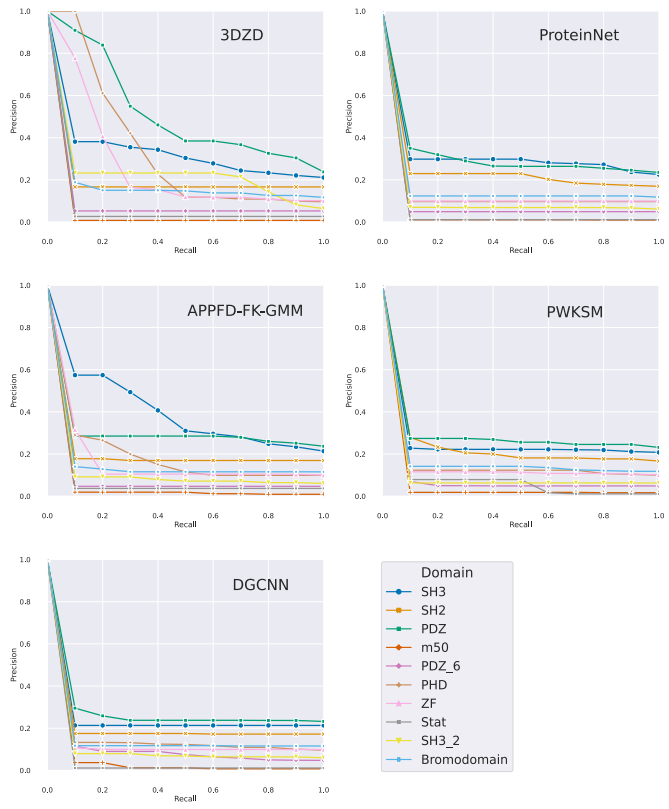


Fig. 7. Per-query precision-recall curves for the shape-only dataset, for each method. All plots are colored according to the legend on the bottom right of the figure.

Table 4. Summary of the average evaluation metrics for the shape-only dataset. The composite classes are excluded from the average; they are presented in Tables C.7 to C.10.

Method	Nearest Neighbor	First Tier	Second Tier	Mean Reciprocal Rank
3DZD	**0.5**	**0.160**	**0.292**	**0.523**
ProteinNet	0	0.088	0.195	0.126
APPFD	0.3	0.136	0.237	0.410
PWKSM	0.1	0.105	0.201	0.236
DGCNN	0	0.098	0.189	0.193

method achieves an overall level of 0.5 for the nearest neighbor metric, 0.160 for the first tier, 0.292 for the second tier and 0.523 for the mean reciprocal rank. These results must be balanced by the fact that a few classes have only a small number of models (namely, the Stat-binding and m50 classes only have 6 and 4 members, respectively, see Figure 2), and thus impact negatively the averaged results. For completeness, Tables C.7, C.8, C.9 and C.10 in Appendix C contain the per-class evaluation metrics for all methods.

The precision-recall curves for each individual classes (Figure 7) show that most methods display a similar behavior for all classes, characterized by a quick drop of the precision at low recall values. A few methods, however, show a different pattern for a few classes (Figure7): see the PDZ class for the 3DZD method (green curve, top left plot) or the SH3 class for the APPFD-FK-GMM method (dark blue curve, middle left plot), for instance, whose corresponding curves display medium precision values at medium recall. The confusion matrices for all methods are shown in Figure 8. Combined with Figure 1, they allow us to put the performance into perspective. For instance, PDZ and PDZ_6 domains are topologically very similar (TM-score: 0.79, Figure 1) and therefore were expected to be confusing. When using the PDZ_6 query, ProteinNet retrieved only 1 (4%) of the 26 PDZ_6 shapes within the first 26 retrieved results, but also 12 (46%) shapes from the PDZ class (Figure 8, middle top confusion matrix). More strikingly, 3DZD only found 1 (3%) of the 33 SH3_2 shapes within the 33 first retrieved shapes using the SH3_2 query, but the other 32 retrieved shapes belong to the SH3 class (Figure 8, second row of the top left confusion matrix), which is closely related to the SH3_2 class (TM-score: 0.84, Figure 1).

5.2. *Shape+electrostatics challenge*

Table 5. Summary of the average evaluation metrics for the shape+electrostatics dataset. The composite classes are excluded from the average; they are presented in Tables D.11 to D.14.

Method	Nearest Neighbor	First Tier	Second Tier	Mean Reciprocal Rank
3DZD	0.5	0.160	0.321	0.454

Similarly to the shape-only dataset, results for the shape+electrostatics dataset are presented in Table 5 and Figures 9 and 10. Only one team returned a dissimilarity matrix for the shape+electrostatics dataset. The evaluation metrics are listed in Table 5. The results show similar trends compared to the shape-only dataset, with a nearest neighbor of 0.5, a first tier value of 0.16, a second tier value of 0.321 and a mean reciprocal rank of 0.454. These metrics are similar to the results obtained from the shape-only dataset for the 3DZD method (the second tier value increased while the mean reciprocal rank decreased). The per-class metrics are shown in Appendix D (Tables D.11, D.12, D.13 and D.14).

The precision-recall curves (Figure 9) show a similar overall behavior for the 3DZD method, whose performance improved significantly for the SH3 domain but decreased significantly for the PDZ domain (dark blue and green curves, respectively, left plot of Figure 9). The confusion matrix (Figure 10) is in line with the previous results, indicating that 3DZD performs similarly in terms of overall performance but with a few differences at the per-class results.

6. Discussion and concluding remarks

6.1. *Shape-only dataset*

Overall, the 3DZD method obtained the best results, in line with the previous tracks on protein shapes, where this group

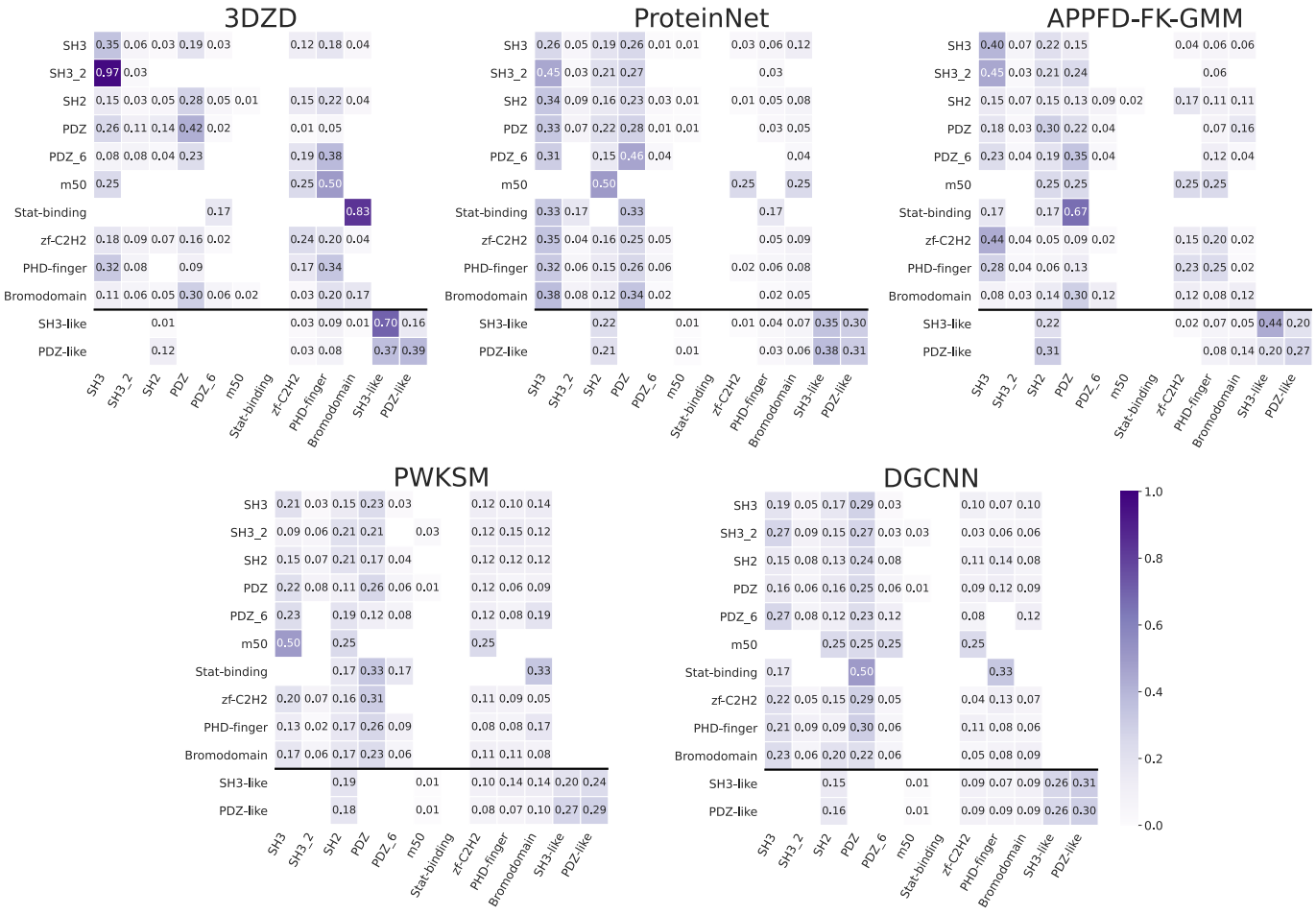


Fig. 8. Confusion matrices of all methods for the shape-only dataset. The color-range is the same for all matrices. Confusion ranges from 0 (white background) to 1 (deep purple background). The original classes are separated from the composite classes (SH3-like and PDZ-like) by a black line.

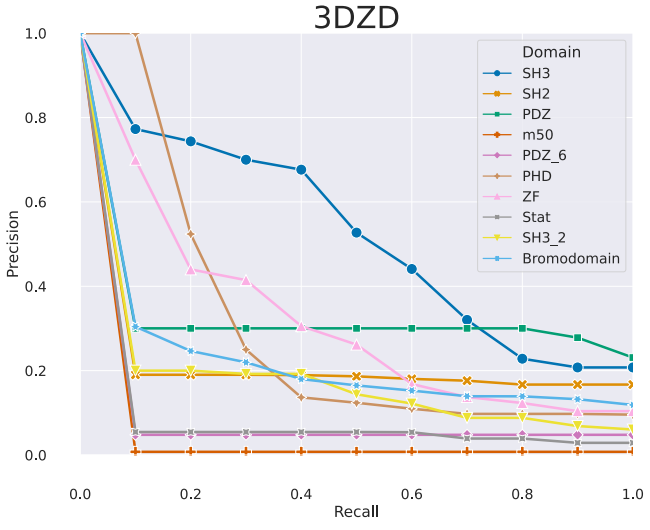


Fig. 9. Per-query precision-recall curves for the shape+electrostatics dataset, for each method. All plots are colored according to the legend on the far right of the figure.

similarly obtained overall good results [33, 34]. This method relies on the use of 3D Zernike polynomials, which has been successfully used to retrieve proteins based on their shapes [9] or their Cα coordinates [13]. It then uses a neural network trained on the SCOPe [37, 38] database, whose classification largely overlaps with the classification of the Pfam database [16]. As instance, in the SCOPe databases, the SH3 domain and the SH3_2 domain are classified in two different SCOPe domains, similarly to the Pfam classification. The DGCNN used the data from another SHREC'21 track, the *Retrieval and classification of protein surfaces equipped with physical & chemical properties* track [35]. The organizers of this track, similarly to a previous track [33], proposed a set of shapes derived from NMR structures along their surficial physico-chemical properties to allow the participants to train their methods, and the resulting classification of the proteins was derived from the SCOPe database as well. The DGCNN and 3DZD methods were therefore trained on similar data, but produced different performance.

Another point to consider, is that the DGCNN method uses a sampling (down to $\approx 8,000$ points) of the initial point clouds (see 3.5.1) that potentially resulted in a loss of information

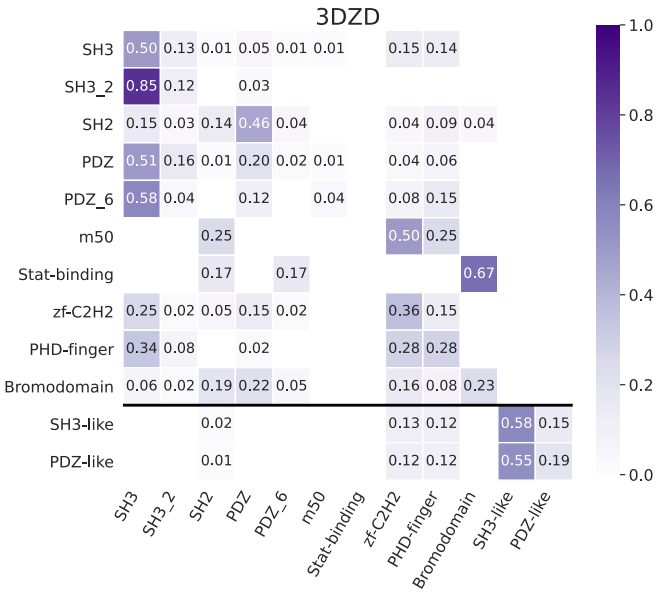


Fig. 10. Confusion matrix for the shape+electrostatics dataset. Confusion ranges from 0 (white background) to 1 (deep purple background). The original classes are separated from the composite classes (SH3-like and PDZ-like) by a black horizontal line.

that might explain the difference of performance between these two groups (DGCNN and 3DZD). However, the ProteinNet and APPFD-FK-GMM methods use more severe down-sampling steps as well to reduce the number of points down to $2,048$ for ProteinNet (see 3.2.2) and to $3,500$ for APPFD-FK-GMM (see 3.3) with various outputs in terms of performance. These numbers should be compared to the initial meshes sizes, which range from $37,658$ to $582,496$ points. The APPFD-FK-GMM group, however, was able to better retrieve relevant results within the first hits, as evidenced by higher values of Nearest-neighbor and Mean Reciprocal Rank for the shape-only dataset (Table 4).

While some methods were able to maintain medium precision levels at medium recall values (see section 5.1), a few queries were difficult to handle for all methods. For the DNA-binding domain from the STAT protein family or the peptidase M50 domain, the low number of such surfaces (6 and 4, respectively) in the datasets explains the low performance observed for all methods. For the other queries, like the PDZ_6 domain or the SH3_2 domain, the explanation is the presence of closely related domains, the PDZ and SH3 domains, respectively. These confusing classes are significantly more populated (128 *versus* 26, for the PDZ / PDZ_6 domains, and 115 *versus* 33, for the SH3 / SH3_2 domains). This is supported by the confusion matrices, which showed that, for instance, the ProteinNet group retrieved a great amount of SH3 domains (and almost no SH3_2 domains) within the top results using the SH3_2 query, or the DGCNN group retrieved a significantly higher proportion of PDZ domains than PDZ_6 domains within the first results using the PDZ_6 query. In these cases, the high level of similarity between the domains coupled to the imbalanced size of the classes have negatively impacted the results. In the mean time, these results highlight the limits of the currently available methods to distinguish between the most closely related proteins.

Also, we observed different results for order-aware (mean reciprocal rank) and order-unaware (nearest neighbor, first ter and second tier) metrics. While DGCNN, ProteinNet and PWKSM methods display similar values for order-unaware metrics, PWKSM displays a higher Mean Reciprocal Rank. When converted back to ranks, these results mean that, on average, PWKSM ranked better the first relevant match, but did find a similar amount of relevant items within the top results.

6.2. *Shape+electrostatic dataset*

While the shape of a protein is of tremendous importance, its surficial properties are important as well. Therefore, in this track, we generated the shape+electrostatic dataset which encompass both properties, in order to stimulate the development of such methods. However, only one groups returned a dissimilarity matrix for this dataset, namely the 3DZD group. Most groups that participated to this challenge come from the computer vision field. As such, most of the methods presented in this work are the result of methodological developments dedicated to the analysis of 3D point clouds. The development of new, specific methods to handle both shapes and electrostatics would require an amount of time far greater than the SHREC timeline. Nevertheless, we hope that this challenge along with the other challenge dedicated to protein shapes [35] would stimulate the development of such methods.

The overall results showed that the treatment of the electrostatics by the 3DZD marginally improved the results, compared to the shape-only results. Interestingly, the electrostatic potential impacted differently each class: it improved the ability of the 3DZD method to handle the SH3 or the Bromodomain classes, as evidenced by the precision-recall curves (Figures 7 and 9, Tables in Appendix C and Appendix D). The 3DZD method is derived from previous attempts from the same team to couple shape and electrostatics analysis to classify protein surfaces [10]. As noted in this exploratory work, electrostatics may be suitable to compare closely related proteins, while our datasets was mainly composed of loosely related proteins (Fig 1). The electrostatics potential feature is likely to be more beneficial for protein surface comparison of local features rather than global shapes. In such local cases (like comparison of catalytic or binding sites), local electrostatic hot spots would represent the major local feature rather than one of the many features of the global protein surface, as it is the case for the MaSIF method [11].

6.3. *Current machine learning–based methods: pitfalls and challenges*

Pitfalls of previously mentioned methods lies in the exploited protein datasets and their characteristics, notably the class imbalance as well as the high inter–class shape similarity. Indeed, the protein datasets are often highly imbalanced in terms of protein classes, which introduces a bias in the training process of these methods towards learning efficiently class representation. One common technique consists of using data augmentation to

overcome this lack of original data. Hence, it is important to bear in mind that several protein classes cannot be considered as representative groups of protein families. Moreover, our problematic tackles a large quantity of classes composed of protein, which are *e.g.* visually highly similar. In such a case, a challenge lies in the design of new methods with a high discriminating power that allows to extract the most significant features for distinguishing between protein classes. In this sense, other aspects of the proteins (in addition to the shape) such as molecular properties and electrostatic properties could be considered. These parameters have to be carefully analyzed through experiments before envisaging method generalizations."

6.4. Concluding remarks

In conclusion, we have presented the results of the SHREC'21 challenge on *Surface-based protein domains retrieval*. The number of participants remained stable compared to the last two years, indicating a constant interest of the shape retrieval community towards biologically relevant problems. Each group relied on different methods and theoretical background with respect to recommended modeling/machine learning practices [63, 64] in order to solve the problem proposed by the organizers, and represent a variety of approaches to the same problem. As a step towards open science, all participants accepted to share their programs publicly with the community. Overall, the results are decreased compared to similar past tracks [34]. Indeed, two methods based on descriptors similar to 3DZD and APPFD-FK-GMM (3DZD and HAPPS, respectively) were presented in the SHREC'20 contest and performed very well (*e.g.* both methods exceeding 0.95 for the NN metric) on a problem similar to the shape-only problem (see Tables 6 and 7 of [34]). However, the adapted versions (3DZD and APPFD-FK-GMM) did not reach the same level of performance by exploiting this new, particular dataset of proteins. This decrease of performance (and low performances from the three other methods) reveals that this year dataset was particularly hard to analyze, and that there is still room for improvements. Among the proposed methods, we observed that 3 over 5 used a learning-based protocol at some point. This proportion is in line with last year track, and show that such approaches continue to be investigated as they usually improve the results. To this regard, the SHREC'21 track on *Retrieval and classification of protein surfaces equipped with physical & chemical properties* might highlight some interesting points on the best architecture to learn protein surficial properties [35]. Similarly, the organizers of this track computed a set of additional chemical properties (electrostatic potential, location of potential hydrogen bond donors and acceptors, hydrophobicity). In this track, the participants first used the surface geometry then the combination of the geometry and physico-chemical features of the protein surfaces. The results showed that all methods improved their results when using both the geometric and physico-chemical data compared the the geometry only. Particularly, the results generated by the machine learning based methods increased more compared to the other methods. As each of the physico-chemical feature was not considered individually, it remains hard to knowpredict whether one feature hasbeat a greater importance than the other. However, in their work, Gainza et al. showed that the electrostatic potential have the greatest impact of the physico-chemical features they computed. In our work, the electrostatic potential was used by the 3DZD team as an additional feature to help the retrieval task. The results reveals only slightly improved results compared to the results from the shape–only dataset. As noted in [10], the electrostatic potential may be of better use to compare closely related proteins, rather than comparing loosely related proteins, as it is the case in our work. Alternatively, shapes and electrostatics may be used in hierarchical way, *i.e.* using first the shape then the electrostatics to achieve a better result.

This track reveals significantly lower results when compared to past tracks [31, 32, 33, 34]. However, satisfactory solutions exist to distinguish between loosely related proteins, or to identify identical proteins with different conformations, based on their shapes only. Our work also reveals some limits of the methods used by the participants for the challenge. Very closely related proteins (such as SH3 and SH3_2 protein domains), *i.e.* proteins displaying a high topological similarity and limited variations of their amino-acid sequences, hence surfaces, are still difficult to separate in different classes, but some methods distinguish them from the other classes. Also, when we consider the DNA-binding domain from the STAT proteins, no method was able to produce satisfactory results. While the DNA-binding domain used as a query has a globular shape, the STAT proteins are significantly bigger, with a non-globular shape, and have 3 additional domains (1 of which is a SH2 domain, a domain included in the dataset), which means that only partial matches compared to the query may be achieved. This specific issue (the comparison of partially overlapping objects) may require further development.

In the future, this latter point could be the subject of a dedicated SHREC track, and a good indicator of the overall progresses made in the field of the retrieval of proteins based on their surfaces. Currently, most methods have difficulties to handle such cases, which are quite common. Solving this challenge would be a step forward for the community. At the same time, explainable artificial intelligence (XAI) methods [65] may highlight the latent features responsible for good or bad predictions, and help decipher the results of machine learning–based methods. XAI methods may help explain the performance difference observed for each class of protein, and provide a human-interpretable representation of machine learning descriptors, and therefore help identifying the current limits of these algorithms. Finally, deciphering to which extend, if any, the standard physico-chemical features (electrostatics potential, charges distribution, hydrophobicity, *etc.*) improve the results may be the main focus of the next SHREC tracks devoted to protein surfaces.

Additional statments & Acknowledgments

The authors thank the 3DOR 2021 Workshop organizing committee for maintaining this workshop despite the current COVID-19 pandemic. Matthieu Montès and Florent Langenfeld thank Taoufik Labib for setting up and maintaining the track webpage.

Funding. Léa Sirugue, Matthieu Montès and Florent Langenfeld are supported by the European Research Council Executive Agency under the research grant number 640283. Daisuke Kihara acknowledges supports from the National Institutes of Health (R01GM133840, R01GM123055) and the National Science Foundation (DBI2003635, CMMI1825941, and MCB1925643). Charles Christoffer is supported by NIGMS-funded pre–doctoral fellowship (T32 GM132024). Huu-Nghia H. Nguyen, Tuan-Duy H. Nguyen, Vinh-Thuyen Nguyen-Truong, Danh Le, Hai-Dang Nguyen, and Minh-Triet Tran are supported by National University Ho Chi Minh City (VNU-HCM) (DS2020-42-01).

Author contributions. Conceptualization: Florent Langenfeld & Matthieu Montès; *Methodology*: Tunde Aderinwale, Charles Christoffer, Woong-Hee Shin, Genki Terashi, Xiao Wang & Daisuke Kihara (3DZD method), Halim Benhabiles, Karim Hammoudi, Adnane Cabani, Feryal Windal & Mahmoud Melkemi (ProteinNet), Ekpo Otu, Reyer Zwiggelaar, David Hunter & Yonghuai Liu (APPFD-FK-GMM), Léa Sirugue & Matthieu Montès (PWKSM), Huu-Nghia H. Nguyen, Tuan-Duy H. Nguyen, Vinh-Thuyen Nguyen-Truong, Danh Le, Hai-Dang Nguyen & Minh-Triet Tran (DGCNN); *Formal analysis and investigation*: Florent Langenfeld & Matthieu Montès; *Writing – original draft preparation*: Florent Langenfeld; *Writing – review and editing*: all authors.

Conflict of interest. The authors have no relevant financial or non-financial interests to disclose.

References

[1] Wolynes, P, Onuchic, J, Thirumalai, D. Navigating the folding routes. Science 1995;267(5204):1619–1620. URL: https://doi.org/10.1126/science.7886447. doi:10.1126/science.7886447.

[2] Karplus, M. Behind the folding funnel diagram. Nature Chemical Biology 2011;7(7):401–404. URL: https://doi.org/10.1038/nchembio.565. doi:10.1038/nchembio.565.

[3] Holm, L, Sander, C. Mapping the protein universe. Science 1996;273(5275):595–602. URL: https://doi.org/10.1126/science.273.5275.595. doi:10.1126/science.273.5275.595.

[4] Shindyalov, IN, Bourne, PE. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Engineering Design and Selection 1998;11(9):739–747. URL: https://doi.org/10.1093/protein/11.9.739. doi:10.1093/protein/11.9.739.

[5] Zemla, A. LGA: A method for finding 3D similarities in protein structures. Nucleic Acids Research 2003;31(13):3370–3374. URL: https://doi.org/10.1093/nar/gkg571. doi:10.1093/nar/gkg571.

[6] Zhang, Y. TM–align: A protein structure alignment algorithm based on the TM–score. Nucleic Acids Research 2005;33(7):2302–2309. URL: https://doi.org/10.1093/nar/gki524. doi:10.1093/nar/gki524.

[7] Mariani, V, Biasini, M, Barbato, A, Schwede, T. lDDT: A local superposition-free score for comparing protein structures and models using distance difference tests. Bioinformatics 2013;29(21):2722–2728. URL: https://doi.org/10.1093/bioinformatics/btt473. doi:10.1093/bioinformatics/btt473.

[8] Shulman-Peleg, A, Nussinov, R, Wolfson, HJ. Recognition of functional sites in protein structures. Journal of Molecular Biology 2004;339(3):607–633. URL: https://doi.org/10.1016/j.jmb.2004.04.012. doi:10.1016/j.jmb.2004.04.012.

[9] Sael, L, Li, B, La, D, Fang, Y, Ramani, K, Rustamov, R, et al. Fast protein tertiary structure retrieval based on global surface shape similarity. Proteins: Structure, Function, and Bioinformatics 2008;72(4):1259–1273. URL: https://doi.org/10.1002/prot.22030. doi:10.1002/prot.22030.

[10] Sael, L, La, D, Li, B, Rustamov, R, Kihara, D. Rapid comparison of properties on protein surface. Proteins: Structure, Function, and Bioinformatics 2008;73(1):1–10. URL: https://doi.org/10.1002/prot.22141. doi:10.1002/prot.22141.

[11] Gainza, P, Sverrisson, F, Monti, F, Rodolà, E, Boscaini, D, Bronstein, MM, et al. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. Nature Methods 2019;17, pages184–192(2020):184–192. URL: http://infoscience.epfl.ch/record/273279. doi:10.1038/s41592-019-0666-6.

[12] Zhang, Y, Sui, X, Stagg, S, Zhang, J. FTIP: An accurate and efficient method for global protein surface comparison. Bioinformatics 2020;36(10):3056–3063. URL: https://doi.org/10.1093/bioinformatics/btaa076. doi:10.1093/bioinformatics/btaa076.

[13] Guzenko, D, Burley, SK, Duarte, JM. Real time structural search of the Protein Data Bank. PLOS Computational Biology 2020;16(7):e1007970. URL: https://doi.org/10.1371/journal.pcbi.1007970. doi:10.1371/journal.pcbi.1007970.

[14] Zhang, Z, Witham, S, Alexov, E. On the role of electrostatics in protein–protein interactions. Physical Biology 2011;8(3):035001. doi:10.1088/1478-3975/8/3/035001.

[15] Zhang, Y, Skolnick, J. Scoring function for automated assessment of protein structure template quality. Proteins: Structure, Function, and Bioinformatics 2004;57(4):702–710. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.20264. doi:https://doi.org/10.1002/prot.20264.

[16] Mistry, J, Chuguransky, S, Williams, L, Qureshi, M, Salazar, G, Sonnhammer, ELL, et al. Pfam: The protein families database in 2021. Nucleic Acids Research 2020;49(D1):D412–D419. URL: https://doi.org/10.1093/nar/gkaa913. doi:10.1093/nar/gkaa913.

[17] Berman, HM, Westbrook, J, Feng, Z, Gilliland, G, Bhat, TN, Weissig, H, et al. The Protein Data Bank. Nucleic Acids Research 2000;28(1):235–242. URL: https://doi.org/10.1093/nar/28.1.235. doi:10.1093/nar/28.1.235.

[18] Berman, H, Henrick, K, Nakamura, H. Announcing the worldwide Protein Data Bank. Nature Structural & Molecular Biology 2003;10(12):980–980. URL: https://doi.org/10.1038/nsb1203-980. doi:10.1038/nsb1203-980.

[19] Takashima, H. High-resolution protein structure determination by NMR. In: Webb, G, editor. Annual Reports on NMR Spectroscopy; vol. 59. Elsevier; 2006, p. 235–273. URL: https://doi.org/10.1016/s0066-4103(06)59005-2. doi:10.1016/s0066-4103(06)59005-2.

[20] Consortium, TU. UniProt: The universal protein knowledgebase in 2021. Nucleic Acids Research 2020;49(D1):D480–D489. URL: https://doi.org/10.1093/nar/gkaa1100. doi:10.1093/nar/gkaa1100.

[21] Dolinsky, TJ, Nielsen, JE, McCammon, JA, Baker, NA. PDB2PQR: An automated pipeline for the setup of Poisson–Boltzmann electrostatics calculations. Nucleic Acids Research 2004;32(Web Server):W665–W667. URL: https://doi.org/10.1093/nar/gkh381. doi:10.1093/nar/gkh381.

[22] Søndergaard, CR, Olsson, MHM, Rostkowski, M, Jensen, JH. Improved treatment of ligands and coupling effects in empirical calculation and rationalization of pKa values. Journal of Chemical Theory and Computation 2011;7(7):2284–2295. URL: https://doi.org/10.1021/ct200133y. doi:10.1021/ct200133y.

[23] Olsson, MHM, Søndergaard, CR, Rostkowski, M, Jensen, JH. PROPKA3: Consistent treatment of internal and surface residues in empirical pKa predictions. Journal of Chemical Theory and Computation 2011;7(2):525–537. URL: https://doi.org/10.1021/ct100578z. doi:10.1021/ct100578z.

[24] Xu, D, Zhang, Y. Generating triangulated macromolecular surfaces by Euclidean distance transform. PLoS ONE 2009;4(12):e8140. URL: https://doi.org/10.1371/journal.pone.0008140. doi:10.1371/journal.pone.0008140.

[25] Xu, D, Li, H, Zhang, Y. Protein depth calculation and the use for improving accuracy of protein fold recognition. Journal of Computational Biology 2013;20(10):805–816. URL: https://doi.org/10.1089/cmb.2013.0071. doi:10.1089/cmb.2013.0071.

[26] Baker, NA, Sept, D, Joseph, S, Holst, MJ, McCammon, JA. Electrostatics of nanosystems: Application to microtubules and the ribosome.

Proceedings of the National Academy of Sciences 2001;98(18):10037–10041. URL: https://doi.org/10.1073/pnas.181342398. doi:10.1073/pnas.181342398.

[27] Söding, J. Protein homology detection by HMM–HMM comparison. Bioinformatics 2004;21(7):951–960. URL: https://doi.org/10.1093/bioinformatics/bti125. doi:10.1093/bioinformatics/bti125.

[28] Xu, J, Zhang, Y. How significant is a protein structure similarity with TM-score = 0.5? Bioinformatics 2010;26(7):889–895. URL: https://doi.org/10.1093/bioinformatics/btq066. doi:10.1093/bioinformatics/btq066.

[29] Temerinac-Ott, M, Reisert, M, Burkhardt, H. SHREC'07 - Protein Retrieval Challenge. 2008.

[30] Mavridis, L, Venkatraman, V, Ritchie, DW, Morikawa, N, Andonov, R, Cornu, A, et al. SHREC'10 Track: Protein Model Classification. In: Daoudi, M, Schreck, T, editors. Eurographics Workshop on 3D Object Retrieval. The Eurographics Association. ISBN 978-3-905674-22-4; 2010, p. 117–124. doi:10.2312/3DOR/3DOR10/117-124.

[31] Song, N, Craciun, D, Christoffer, CW, Han, X, Kihara, D, Levieux, G, et al. Protein shape retrieval. In: Pratikakis, I, Dupont, F, Ovsjanikov, M, editors. Eurographics Workshop on 3D Object Retrieval. The Eurographics Association. ISBN 978-3-03868-030-7; 2017, p. 67–74. URL: https://diglib.eg.org/handle/10.2312/3dor20171055. doi:10.2312/3DOR.20171055.

[32] Langenfeld, F, Axenopoulos, A, Chatzitofis, A, Craciun, D, Daras, P, Du, B, et al. Protein Shape Retrieval. In: Telea, A, Theoharis, T, Veltkamp, R, editors. Eurographics Workshop on 3D Object Retrieval. The Eurographics Association. ISBN 978-3-03868-053-6; 2018, p. 53–61. doi:10.2312/3dor.20181053.

[33] Langenfeld, F, Axenopoulos, A, Benhabiles, H, Daras, P, Giachetti, A, Han, X, et al. Protein Shape Retrieval Contest. In: Biasotti, S, Lavoué, G, Veltkamp, R, editors. Eurographics Workshop on 3D Object Retrieval. The Eurographics Association. ISBN 978-3-03868-077-2; 2019, p. 25–31. doi:10.2312/3dor.20191058.

[34] Langenfeld, F, Peng, Y, Lai, YK, Rosin, PL, Aderinwale, T, Terashi, G, et al. SHREC 2020: Multi-domain protein shape retrieval challenge. Computers & Graphics 2020;91:189–198. URL: https://doi.org/10.1016/j.cag.2020.07.013. doi:10.1016/j.cag.2020.07.013.

[35] Raffo, A, Fugacci, U, Biasotti, S, Rocchia, W, Liu, Y, Otu, E, et al. SHREC 2021 track: Retrieval and classification of protein surfaces equipped with physical and chemical properties. Computers & Graphics 2021;99:1–21. doi:https://doi.org/10.1016/j.cag.2021.06.010.

[36] Canterakis, N. 3D Zernike moments and Zernike affine invariants for 3D image analysis and recognition. In: In 11th Scandinavian Conf. on Image Analysis. 1999, p. 85–93.

[37] Fox, NK, Brenner, SE, Chandonia, JM. SCOPe: Structural classification of proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. Nucleic Acids Research 2013;42(D1):D304–D309. URL: https://doi.org/10.1093/nar/gkt1240. doi:10.1093/nar/gkt1240.

[38] Chandonia, JM, Fox, NK, Brenner, SE. SCOPe: Classification of large macromolecular structures in the structural classification of proteins—extended database. Nucleic Acids Research 2018;47(D1):D475–D481. URL: https://doi.org/10.1093/nar/gky1134. doi:10.1093/nar/gky1134.

[39] Esquivel-Rodríguez, J, Xiong, Y, Han, X, Guang, S, Christoffer, C, Kihara, D. Navigating 3D electron microscopy maps with EM-SURFER. BMC Bioinformatics 2015;16(1). URL: https://doi.org/10.1186/s12859-015-0580-6. doi:10.1186/s12859-015-0580-6.

[40] Qi, CR, Su, H, Mo, K, Guibas, LJ. PointNet: Deep learning on point sets for 3D classification and segmentation. arXiv preprint arXiv:161200593 2016;.

[41] Yuksel, C. Sample elimination for generating Poisson disk sample sets. Computer Graphics Forum (Proceedings of EUROGRAPHICS 2015) 2015;34(2):25–32. URL: http://dx.doi.org/10.1111/cgf.12538. doi:10.1111/cgf.12538.

[42] Benhabiles, H, Hammoudi, K, Windal, F, Melkemi, M, Cabani, A. A transfer learning exploited for indexing protein structures from 3D point clouds. In: Processing and Analysis of Biomedical Information. Springer International Publishing; 2019, p. 82–89. URL: https://doi.org/10.1007/978-3-030-13835-6_10. doi:10.1007/978-3-030-13835-6_10.

[43] Otu, E, Zwiggelaar, R, Hunter, D, Liu, Y. Nonrigid 3D shape retrieval with happs: A novel hybrid augmented point pair signature. In: 2019 International Conference on Computational Science and Computational Intelligence (CSCI). 2019, p. 662–668. doi:10.1109/CSCI49370.2019.00124.

[44] Otu, E. Code Implementation of Agglomeration of Point-pair Feature Descritor With Fisher Kernel and Gaussian Mixture Model (APPFD-FK-GMM). https://github.com/KoksiHub/APPFD_FK_GMM-Method-For-SHREC-2021-Surface-based-Protein-Domains-Retrieval ???? Accessed: 2021-07-09.

[45] Moscoso Thompson, E, Biasotti, S, Giachetti, A, Tortorici, C, Werghi, N, Obeid, AS, et al. SHREC'20 track: Retrieval of digital surfaces with similar geometric reliefs. Computers & Graphics 2020;.

[46] Wahl, E, Hillenbrand, U, Hirzinger, G. Surflet-pair-relation histograms: A statistical 3d-shape representation for rapid classification. In: Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. 2003, p. 474–481.

[47] Aubry, M, Schlickewei, U, Cremers, D. The wave kernel signature: A quantum mechanical approach to shape analysis. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops). 2011, p. 1626–1633. doi:10.1109/ICCVW.2011.6130444.

[48] Rodolà, E, Rota Bulo, S, Windheuser, T, Vestner, M, Cremers, D. Dense non-rigid shape correspondence using random forests. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014, p. 4177–4184.

[49] Boscaini, D, Masci, J, Melzi, S, Bronstein, MM, Castellani, U, Vandergheynst, P. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. Computer Graphics Forum 2015;34(5):13–23. doi:https://doi.org/10.1111/cgf.12693.

[50] Limberger, FA, Wilson, RC. Feature encoding of spectral signatures for 3D non-rigid shape retrieval. In: BMVC. 2015, p. 56–1.

[51] Zeng, H, Liu, Y, Li, S, Che, J, Wang, X. Convolutional neural network based multi-feature fusion for non-rigid 3D model retrieval. Journal of Information Processing Systems 2018;14(1):176–190.

[52] Angenent, S, Haker, S, Tannenbaum, A, Kikinis, R. On the Laplace–Beltrami operator and brain surface flattening. IEEE Transactions on Medical Imaging 1999;18(8):700–711.

[53] Craciun, D, Levieux, G, Montes, M. Shape Similarity System driven by Digital Elevation Models for Non-rigid Shape Retrieval. In: Pratikakis, I, Dupont, F, Ovsjanikov, M, editors. Eurographics Workshop on 3D Object Retrieval. The Eurographics Association. ISBN 978-3-03868-030-7; 2017, p. 51–54. doi:10.2312/3dor.20171051.

[54] Fortune, S, Wyllie, J. Parallelism in random access machines. In: Proceedings of the tenth annual ACM symposium on Theory of computing. 1978, p. 114–118.

[55] Cole, R, Vishkin, U. Faster optimal parallel prefix sums and list ranking. Information and computation 1989;81(3):334–352.

[56] Santos, EE. Optimal and efficient algorithms for summing and prefix summing on parallel machines. Journal of Parallel and Distributed Computing 2002;62(4):517–543.

[57] Wang, Y, Sun, Y, Liu, Z, Sarma, SE, Bronstein, MM, Solomon, JM. Dynamic graph CNN for learning on point clouds. ACM Transactions On Graphics 2019;38(5):1–12.

[58] van Rossum, G, Python Development Team, . Python 3.8.8 documentation. 2021. URL: https://docs.python.org/3.8/.

[59] Paszke, A, Gross, S, Massa, F, Lerer, A, Bradbury, J, Chanan, G, et al. Pytorch: An imperative style, high-performance deep learning library. arXiv preprint arXiv:191201703 2019;.

[60] Fey, M, Lenssen, JE. Fast graph representation learning with pytorch geometric. arXiv preprint arXiv:190302428 2019;.

[61] Moscoso Thompson, E, Biasotti, S, Giachetti, A, Tortorici, C, Werghi, N, Obeid, AS, et al. SHREC 2020: Retrieval of digital surfaces with similar geometric reliefs. Computers & Graphics 2020;91:199–218. URL: https://doi.org/10.1016/j.cag.2020.07.011. doi:10.1016/j.cag.2020.07.011.

[62] Shilane, P, Min, P, Kazhdan, M, Funkhouser, T. The Princeton Shape Benchmark. In: Proceedings of the Shape Modeling International 2004. SMI '04; USA: IEEE Computer Society. ISBN 0769520758; 2004, p. 167–178.

[63] Caruana, R, Lundberg, S, Ribeiro, MT, Nori, H, Jenkins, S. Intelligible and explainable machine learning: Best practices and practical chal-

lenges. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. KDD '20; New York, NY, USA: Association for Computing Machinery. ISBN 9781450379984; 2020, p. 3511–3512. doi:10.1145/3394486.3406707.

[64] Artrith, N, Butler, KT, Coudert, FX, Han, S, Isayev, O, Jain, A, et al. Best practices in machine learning for chemistry. Nature Chemistry 2021;13(6):505–508. URL: https://doi.org/10.1038/s41557-021-00716-z. doi:10.1038/s41557-021-00716-z.

[65] Das, A, Rad, P. Opportunities and challenges in explainable artificial intelligence (XAI): A survey. CoRR 2020;abs/2006.11371. URL: https://arxiv.org/abs/2006.11371. arXiv:2006.11371.

[66] Sonnenburg, ED, Bilwes, A, Hunter, T, Noel, JP. The structure of the membrane distal phosphatase domain of RPTPα reveals interdomain flexibility and an SH2 domain interaction region. Biochemistry 2003;42(26):7904–7914. URL: https://doi.org/10.1021/bi0340503. doi:10.1021/bi0340503. arXiv:https://doi.org/10.1021/bi0340503; pMID: 12834342.

[67] Sonnenburg, E, Bilwes, A, Hunter, T, Noel, J. Crystal structure of the Src SH2 domain complexed with peptide (SDpYANFK). 2003. URL: https://doi.org/10.2210/pdb1p13/pdb. doi:10.2210/pdb1p13/pdb.

[68] Musacchio, A, Saraste, M, Wilmanns, M. High-resolution crystal structures of tyrosine kinase SH3 domains complexed with proline-rich peptides. Nature Structural Biology 1994;1(8):546–551. URL: https://doi.org/10.1038/nsb0894-546. doi:10.1038/nsb0894-546.

[69] Musacchio, A, Wilmanns, M, Saraste, M. Crystal structure of the complex of the Abl tyrosine kinase SH3 domain with 3BP-1 synthetic peptide. 1995. URL: https://doi.org/10.2210/pdb1abo/pdb. doi:10.2210/pdb1abo/pdb.

[70] Ponna, SK, Myllykoski, M, Boeckers, TM, Kursula, P. Structure of an unconventional SH3 domain from the postsynaptic density protein Shank3 at ultrahigh resolution. Biochemical and Biophysical Research Communications 2017;490(3):806–812. URL: https://doi.org/10.1016/j.bbrc.2017.06.121. doi:10.1016/j.bbrc.2017.06.121.

[71] Ponna, S, Myllykoski, M, Boeckers, T, Kursula, P. Unconventional SH3 domain from the postsynaptic density scaffold protein Shank3. 2017. URL: https://doi.org/10.2210/pdb5o99/pdb. doi:10.2210/pdb5o99/pdb.

[72] Elkins, JM, Papagrigoriou, E, Berridge, G, Yang, X, Phillips, C, Gileadi, C, et al. Structure of PICK1 and other PDZ domains obtained with the help of self-binding C–terminal extensions. Protein Science 2007;16(4):683–694. URL: https://doi.org/10.1110/ps.062657507. doi:10.1110/ps.062657507.

[73] Faucher, F, de Jesus-Tran, KP, Cantin, L, Luu-the, V, Labrie, F, Breton, R. Crystal structure of 17alpha-hydroxysteroid dehydrogenase in binary complex with NADP(H) in an open conformation. 2006. URL: https://doi.org/10.2210/pdb2he5/pdb. doi:10.2210/pdb2he5/pdb.

[74] Roos, A, Elkins, J, Savitsky, P, Wang, J, Ugochukwu, E, Murray, J, et al. The crystal structure of the PDZ domain of human Microtubule Associated Serine/Threonine Kinase 3 (MAST3). 2009. URL: https://doi.org/10.2210/pdb3khf/pdb. doi:10.2210/pdb3khf/pdb.

[75] Feng, L, Yan, H, Wu, Z, Yan, N, Wang, Z, Jeffrey, PD, et al. Structure of a site-2 protease family intramembrane metalloprotease. Science 2007;318(5856):1608–1612. URL: https://doi.org/10.1126/science.1150755. doi:10.1126/science.1150755.

[76] Dong, A, Lin, L, Bountra, C, Arrowsmith, C, Edwards, A, and, RH. Crystal structure of Cryptosporidium parvum bromodomain cgd2_2690. 2018. URL: https://doi.org/10.2210/pdb6cw0/pdb. doi:10.2210/pdb6cw0/pdb.

[77] Horton, JR, Upadhyay, AK, Qi, HH, Zhang, X, Shi, Y, Cheng, X. Enzymatic and structural insights for substrate specificity of a family of Jumonji histone lysine demethylases. Nature Structural & Molecular Biology 2009;17(1):38–43. URL: https://doi.org/10.1038/nsmb.1753. doi:10.1038/nsmb.1753.

[78] Horton, J, Upadhyay, A, Qi, H, Zhang, X, Shi, Y, Cheng, X. Structure of KIAA1718, human Jumonji demethylase, in complex with N-oxalylglycine. 2009. URL: https://doi.org/10.2210/pdb3kv5/pdb. doi:10.2210/pdb3kv5/pdb.

[79] Zhang, X, Glunz, PW, Jiang, W, Schmitt, A, Newman, M, Barbera, FA, et al. Design and synthesis of bicyclic pyrazinone and pyrimidinone amides as potent TF–FVIIa inhibitors. Bioorganic & Medicinal Chemistry Letters 2013;23(6):1604–1607. URL: https://doi.org/10.1016/j.bmcl.2013.01.094. doi:10.1016/j.bmcl.2013.01.094.

[80] Wei, A. Structure of FACTOR VIIA in complex with the inhibitor (6s)-n-(4-CARBAMIMIDOYLBENZYL)-1-CHLORO-3-(CYCLOBUTYLAMINO)-8, 8-DIETHYL-4-OXO-4, 6, 7, 8-TETRAHYDROPYRROLO[1, 2-a]PYRAZINE-6-CARBOXAMIDE. 2013. URL: https://doi.org/10.2210/pdb4isi/pdb. doi:10.2210/pdb4isi/pdb.

[81] Li, J, Rodriguez, JP, Niu, F, Pu, M, Wang, J, Hung, LW, et al. Structural basis for DNA recognition by STAT6. Proceedings of the National Academy of Sciences 2016;113(46):13015–13020. URL: https://doi.org/10.1073/pnas.1611228113. doi:10.1073/pnas.1611228113.

[82] Li, J, Niu, F, Ouyang, S, Liu, Z. Transcription factor-DNA complex. 2016. URL: https://doi.org/10.2210/pdb5d39/pdb. doi:10.2210/pdb5d39/pdb.

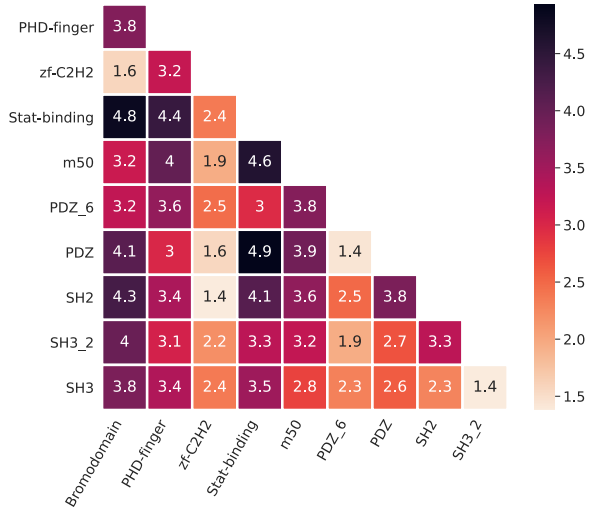


Fig. B.11. Cα-RMSD (Root Mean Square Deviations) between queries structures. The higher the RMSD, the more distant the structures.

Appendix A. List of PDB structures used as queries for the dataset

Appendix B. RMSD between queries structures.

Appendix C. Evaluation metrics details for the shape-only dataset: reciprocal rank, per-class nearest-neighbor, first tier and second tier.

Appendix D. Evaluation metrics details for the shape+electrostatics dataset: reciprocal rank, per-class nearest-neighbor, first tier and second tier.

Table A.6. List of the Protein Data Bank [17, 18] structures used as queries for the track.

Domain name	Pfam ID	PDB code	chain	residues	Reference
SH2 domain	PF00017	1P13	B	161-243	[66, 67]
SH3 domain	PF00018	1ABO	B	67-113	[68, 69]
Variant SH3 domain (SH3_2)	PF07653	5O99	B	474-527	[70, 71]
PDZ domain	PF00595	2HE2	B	421-499	[72, 73]
PDZ_6 domain	PF17820	3KHF	B	982-1034	[74]
Peptidase family M50 (m50)	PF02163	3B4R	B	111-186	[75]
Bromodomain	PF00439	6CW0	B	10-95	[76]
PHD-finger domain	PF00628	3KV5	D	39-88	[77, 78]
Zinc-finger domain, C2H2 type (zf-C2H2)	PF00096	4ISI	D	472-493	[79, 80]
STAT protein, DNA-binding domain (Stat-binding)	PF02864	5D39	D	277-413	[81, 82]

Table C.7. Per-class nearest-neighbor for the shape-only dataset.

Method Class size	SH3 115	SH3_2 33	SH2 92	PDZ 128	PDZ_6 26	m50 4	STAT 6	zf-C2H2 55	PHD 53	Bromodomain 64	SH3-like 148	PDZ-like 154
3DZD	0	0	1	1	0	0	0	1	1	1	1	1
ProteinNet	0	0	0	0	0	0	0	0	0	0	1	0
APPFD	0	0	0	1	0	0	0	1	1	0	0	1
PWKSM	0	0	0	1	0	0	0	0	0	0	0	1
DGCNN	0	0	0	0	0	0	0	0	0	0	0	0

Table C.8. Reciprocal rank for the shape-only dataset.

Method Class size	SH3 115	SH3_2 33	SH2 92	PDZ 128	PDZ_6 26	m50 4	STAT 6	zf-C2H2 55	PHD 53	Bromodomain 64	SH3-like 148	PDZ-like 154
3DZD	0.143	0.033	1	1	0.030	0.002	0.021	1	1	1	1	1
ProteinNet	0.071	0.036	0.25	0.5	0.25	0.005	0.002	0.010	0.071	0.067	1	0.5
APPFD	0.5	0.042	0.1	1	0.2	0.01	0.022	1	1	0.167	0.333	1
PWKSM	0.333	0.053	0.125	1	0.333	0.015	0.042	0.083	0.042	0.333	0.333	1
DGCNN	0.333	0.083	0.167	0.5	0.333	0.037	0.006	0.042	0.1	0.333	0.25	0.5

Table C.9. Per-class first tier for the shape-only dataset.

Method Class size	SH3 115	SH3_2 33	SH2 92	PDZ 128	PDZ_6 26	m50 4	STAT 6	zf-C2H2 55	PHD 53	Bromodomain 64	SH3-like 148	PDZ-like 154
3DZD	0.35	0.03	0.05	0.42	0.00	0.00	0.00	0.24	0.34	0.17	0.70	0.39
ProteinNet	0.26	0.03	0.16	0.28	0.04	0.00	0.00	0.00	0.06	0.05	0.35	0.31
APPFD	0.40	0.03	0.15	0.22	0.04	0.00	0.00	0.15	0.25	0.13	0.44	0.27
PWKSM	0.18	0.06	0.21	0.26	0.08	0.00	0.00	0.11	0.08	0.08	0.20	0.29
DGCNN	0.19	0.09	0.13	0.25	0.12	0.00	0.00	0.04	0.08	0.09	0.26	0.30

Table C.10. Per-class second tier for the shape-only dataset.

Method Class size	SH3 115	SH3_2 33	SH2 92	PDZ 128	PDZ_6 26	m50 4	STAT 6	zf-C2H2 55	PHD 53	Bromodomain 64	SH3-like 148	PDZ-like 154
3DZD	0.55	0.45	0.13	0.72	0.04	0.00	0.00	0.33	0.45	0.27	0.79	0.69
ProteinNet	0.55	0.09	0.43	0.51	0.08	0.00	0.00	0.02	0.15	0.13	0.68	0.60
APPFD	0.58	0.12	0.30	0.53	0.08	0.00	0.00	0.18	0.34	0.23	0.66	0.59
PWKSM	0.39	0.12	0.39	0.49	0.12	0.00	0.00	0.18	0.11	0.20	0.46	0.56
DGCNN	0.37	0.09	0.33	0.39	0.12	0.00	0.00	0.20	0.25	0.16	0.49	0.52

Table D.11. Per-class nearest-neighbor for the shape+electrostatics dataset.

Method	SH3	SH3_2	SH2	PDZ	PDZ_6	m50	STAT	zf-C2H2	PHD	Bromodomain	SH3-like	PDZ-like
Class size	115	33	92	128	26	4	6	55	53	64	148	154
3DZD	0	0	0	1	0	0	0	1	1	0	1	0

Table D.12. Reciprocal Rank for the shape+electrostatics dataset.

Method	SH3	SH3_2	SH2	PDZ	PDZ_6	m50	STAT	zf-C2H2	PHD	Bromodomain	SH3-like	PDZ-like
Class size	115	33	92	128	26	4	6	55	53	64	148	154
3DZD	0.5	0.167	0.333	1	0.010	0.003	0.030	1	1	0.5	1	0.5

Table D.13. Per-class first-tier for the shape+electrostatics dataset.

Method	SH3	SH3_2	SH2	PDZ	PDZ_6	m50	STAT	zf-C2H2	PHD	Bromodomain	SH3-like	PDZ-like
Class size	115	33	92	128	26	4	6	55	53	64	148	154
3DZD	0.50	0.12	0.14	0.20	0.00	0.00	0.00	0.36	0.28	0.23	0.58	0.19

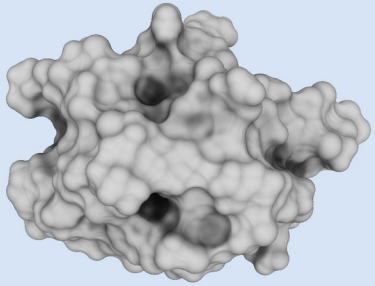
Table D.14. Per-class second tier for the shape+electrostatics dataset.

Method	SH3	SH3_2	SH2	PDZ	PDZ_6	m50	STAT	zf-C2H2	PHD	Bromodomain	SH3-like	PDZ-like
Class size	115	33	92	128	26	4	6	55	53	64	148	154
3DZD	0.68	0.36	0.38	0.55	0.00	0.00	0.00	0.51	0.36	0.36	0.74	0.74

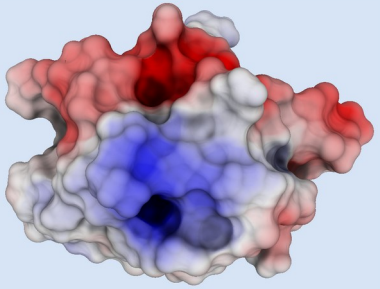
Shape Retrieval Contest 2021 On Protein Surfaces

Queries : 10 proteins

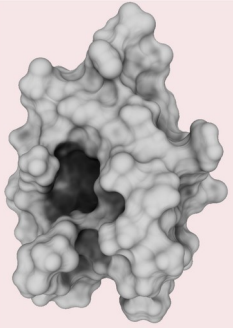
Datasets : 554 proteins



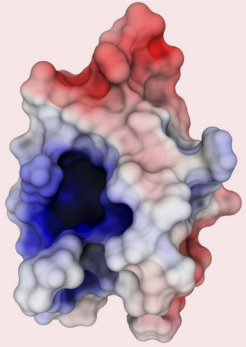
Shape only



Shape + electrostatics



Shape only



Shape + electrostatics