A Transfer Learning Exploited for Indexing Protein Structures from 3D Point Clouds

Halim Benhabiles[1(✉)], Karim Hammoudi[2,3], Feryal Windal[1], Mahmoud Melkemi[2,3], and Adnane Cabani[4]

[1] ISEN-Lille, Yncréa Hauts-de-France, Lille, France
{halim.benhabiles,feryal.windal}@yncrea.fr
[2] Department of Computer Science, IRIMAS, Université de Haute-Alsace, 68100 Mulhouse, France
[3] Université de Strasbourg, Strasbourg, France
{karim.hammoudi,mahmoud.melkemi}@uha.fr
[4] Normandie University, UNIROUEN, ESIGELEC, IRSEEM, 76000 Rouen, France
adnane.cabani@esigelec.fr

Abstract. In this paper, we propose a transfer learning-based methodology that can be exploited for indexing protein structures from associated 3D point clouds. Such a methodology can be particularly useful for biologists that are searching automated solutions to find family members of a query protein or even to label new structures by directly using input raw 3D point clouds. Comparative study and performance evaluation show the efficiency and the potential of the proposed methodology.

Keywords: Transfer learning · Protein structure analysis · Indexing 3D point clouds · PDB · Biomedical imaging

1 Introduction and Motivation

Identifying protein functions and analyzing their interactions can help to understand the mechanisms that govern the living beings, and accordingly, to establish new effective therapeutic strategies. In most cases, functions of a protein can be predicted through analysis of its structure, itself characterized by the composition of its molecules (e.g., amino acids) as well as their relationships and spatial positions [4].

In this sense, methods are used for separating proteins from their other cellular compounds (e.g., ultracentrifugation, electrophoresis). Then, their structures can be studied by varied methods such as X-ray crystallography, Nuclear Magnetic Resonance or mass spectrometer. Biologists and biochemists from around the world regularly exploit these analysis methods and submit their obtained data (e.g., 3D structural information of biological macromolecules) in a mutual and public database that is named Protein Data Bank (PDB[1]) [2].

[1] Guide to Understanding PDB Data: https://pdb101.rcsb.org/learn/guide-to-understanding-pdb-data/introduction.

N. Lepore et al. (Eds.): SaMBa 2018, LNCS 11379, pp. 82–89, 2019.
https://doi.org/10.1007/978-3-030-13835-6_10

Various bioinformatics research topics that have been investigated in the literature for analyzing proteins are presented hereafter.

Due to the increasing interest for the analysis of protein and to the development of emerging instruments and technologies, the size and the diversity of digitized protein information are more and more high *making then complex the exploitation for such a database*. In [5], a freely available web-based database exploration tool (PDB-Explorer[2] website) is proposed and permits to interactively visualize and explore the structural diversity of the PDB (e.g., through color-coded map generation or structure classification).

In [14], the author tackles the *problem of functional annotation* from protein 3D structures for which most solutions use 3D structure superposition techniques that are computationally demanding. The author combines geometry characteristics and physicochemical features for efficiently analyzing the protein surfaces.

In [7], the authors study the *problem of understanding protein-protein interactions*. They propose a methodology of predicting of Hot-Spots in protein-protein interfaces. The presented model is trained on a large number of structural and evolutionary sequence-based features. Also, several classification algorithms with cost functions are utilized. The best model is selected by using c-forest, a random forest ensemble learning method.

In this paper, our goal is to present a transfer learning-based methodology for indexing protein structures represented by 3D point clouds. Indeed, a neural network training process can be computationally time consuming. Additionally, it requires the preparation of ground-truths which is a fastidious task (manual data labelling). Hence, instead of training a neural network, a pre-trained one with generic 3D objects is directly exploited to characterise protein structures. Our proposed indexing methodology is important for biologists that are searching automated solutions to find family members of a query protein or even to label new structures by directly using input raw 3D point clouds.

2 Proposed Methodology

A transfer learning is an operation that consists of exploiting knowledge gained to solve a problem and applying it to solve a different but related problem. Nevertheless, efficient transfer learning needs surrounding processing stages for its adaptation to the targeted problem with respect to its applicative context. In this section, we describe the proposed methodology which is entitled "Generic Learning-based Transfer for Indexing Proteins (GLT4IP)". It is focused on a transfer learning-based indexing method for 3D protein shape retrieval.

Figure 1 provides an overview of the associated major stages. First, the input protein which is represented in the form of a 3D point cloud is resampled and normalized. The resulting pre-processed protein data is injected into a Convolutional Neural Network (CNN) through a classification architecture that was already pre-trained onto a 3D object database. Since this database was composed

[2] PDB-Explorer website: http://www.cheminfo.org/pdbexplorer/.

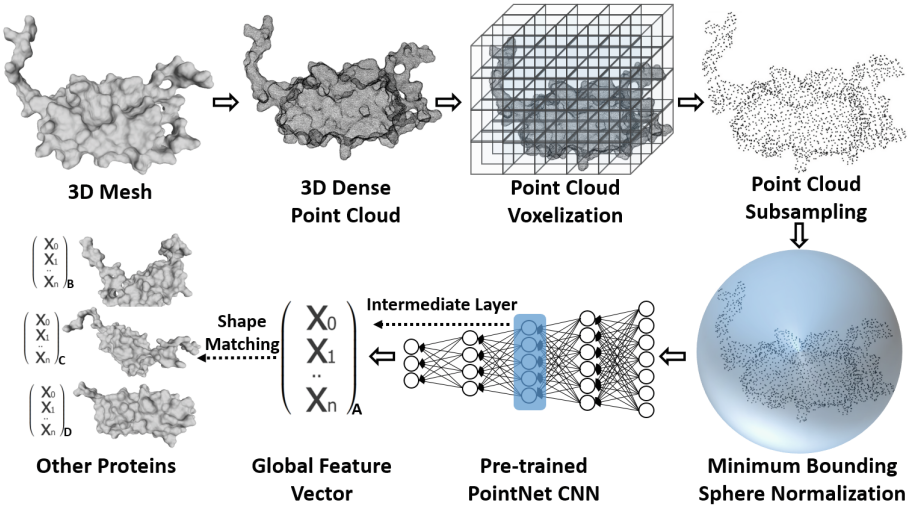

Fig. 1. Overview of our proposed transfer learning-based method.

of a large variety of man-made objects, it made data structures and parameters of the exploited CNN architecture (e.g., associated layers, weight coefficients) particularly tuned for classifying a large variety of object shapes. A transfer learning is then applied by extracting from this CNN architecture, for each protein, a feature vector that is globally embedding structural information of the protein with a generic manner. Finally, extracted protein feature vectors are used to compute the similarity scores from the ones to the others. A sorting of similarity scores can then permit to identify proteins having similar structural characteristics to a query protein—protein shape indexing.

2.1 Sub-sampling of the Considered Protein Point Clouds

Before to proceed to the feature extraction and in order to be able to exploit the considered CNN architecture, the 3D point cloud representing the protein surface (several thousand of points) is sub-sampled in order to reduce its size to 2048 3D points while keeping its global structure. This sub-sampling stage is done to adjust the protein data size to the size of input data that is managed by the CNN architecture. To this end, we apply a volumetric-based clustering algorithm on the original protein by exploiting a simplification method that was proposed in [1]. In particular, the minimum bounding box of the object is subdivided into a 3D voxel grid according to a leaf size parameter (voxel size). This latter parameter is set according to the targeted size of the final point cloud (2048 3D values). The resulting point cloud is then generated by calculating the centroids of the voxels containing points. The main advantage of such a transformation is its ability to preserve the global structure of the object thanks to a uniform sampling of the original surface. Additionally, it is known to be computationally

fast thanks to the use of advanced data structures (see octree of the Point Cloud Library [11]).

2.2 Normalization of the Sub-sampled Protein Point Clouds

Once we obtained the sub-sampled point clouds, the next stage consists of their normalization in order to make coherent the targeted protein-to-protein comparison process. The applied normalization stage is twofold: (i) the sub-sampled 3D point clouds of proteins are spatially rescaled. To reach this goal, the object is normalized into a unit sphere corresponding to the minimal bounding sphere. This step is performed by using an algorithm which has the advantage of not being time consuming ([13] and [9]), (ii) each resulting rescaled 3D point cloud is then re-centered by computing its barycenter and by operating a zero-mean translation to its associated points (i.e. registration of the 3D points to a zero point of common XYZ referential). It is worth mentioning that the quantity of each normalized 3D protein point cloud has not changed and is still equal to 2048.

2.3 Extraction of Structural Feature Vectors

Each prepared protein 3D point cloud (natural 3D object) is then injected into a CNN architecture that was pretrained over a large database of diverse man-made 3D objects in order to benefit from a deep analyzer already calibrated with structural classification objectives (transfer learning). Indeed, deep learning architecture of these recent years are pushing the frontier of performance in many computer vision and 3D applications including data detection, segmentation and classification. Our methodology exploits the PointNet classification architecture [8] as a generic feature vector extractor.

More precisely, in our case we did not consider the output of the last layer of this architecture (i.e. classification vector). We use the pretrained network for extracting a global descriptor vector corresponding to an intermediate fully connected layer giving the best experimental performance. To reach this goal, we have conducted an empirical study to identify which layer level gives the highest performance (see the architecture layers in Fig. 2 of [8]). Consequently, the feature vectors that are generated for the prepared protein implicitly take advantage of information learned on a dataset of approximately 12,300 CAD 3D objects with 40 possible categories (details of operations and training protocols are presented in the PointNet reference).

2.4 Shape Matching

Having generated a descriptor vector for each protein, the last stage consists of measuring the protein-to-protein similarity. To this end, we experimented cost functions over the descriptor vectors, namely the Euclidean distance and the Earth Movers distance [10]. Proteins are sorted from the closest one to the

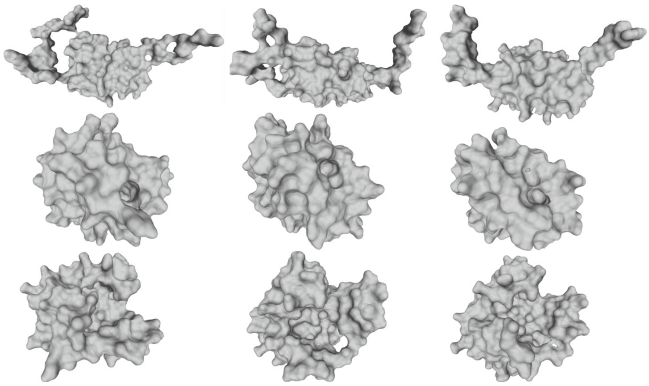

Fig. 2. On each row, examples of proteins belonging to the same class from SHREC2018 protein dataset.

furthest one with respect to each query protein (e.g.; for generating a distance matrix necessary to the object indexing). Both functions provide a dissimilarity score between two compared proteins and a 0 value output means that they are equal.

3 Experimental Results and Performance Evaluation

Our method has been experimented on the SHREC2018 protein dataset and compared to the related state-of-the-art methods [6]. The SHREC2018 protein dataset is composed of 2267 proteins. Each protein is represented by two formats, namely PDB and OFF which give a total number of 4534 files. As raised in the introduction, the PDB (Protein Data Bank) is the standard format that is used by the biologist community. This format describes the protein structure in the form of a point cloud where each point is the center of an atom. The OFF (Object File Format) format describes the surface of the protein in the form of a mesh of triangles. In this latter case, each atom is approximated by a sphere.

The 2267 proteins have been organized into 107 classes where each class represents a protein domain. The dataset has been built following a specific protocol while considering standard references including the protein structure database PDB [2] as well as the SCOPe database (Structural Classification Of Proteins - extended) [3]. For more details on the protocol followed to build the dataset, we refer the reader to the original paper [6]. Figure 2 illustrates some proteins in the OFF format. Each row shows examples of proteins belonging to the same class.

To evaluate the performance of our method, we considered the OFF files of the 2267 proteins. For each protein, we have applied the processing pipeline described in our methodology to extract the feature vectors. As stated previously in the paper, for the feature extraction stage, we employ a transfer learning from the PointNet [8] CNN classification architecture. This allowed to generate

for each protein three feature vectors corresponding to three intermediate and successively fully connected layers for which the sizes are 1024, 512 and 256, respectively.

Figure 3 shows the precision-recall curves obtained by our method for the three feature vectors and using two different distances for the shape matching step: the Euclidean distance and the Earth Movers distance. For this later, we only display the best curve obtained among the three (the one based on a vector of size 1024) for clarity's sake. The figure clearly shows that the best retrieval results correspond to the ones calculated from feature vectors of size 1024 using Euclidean distance.

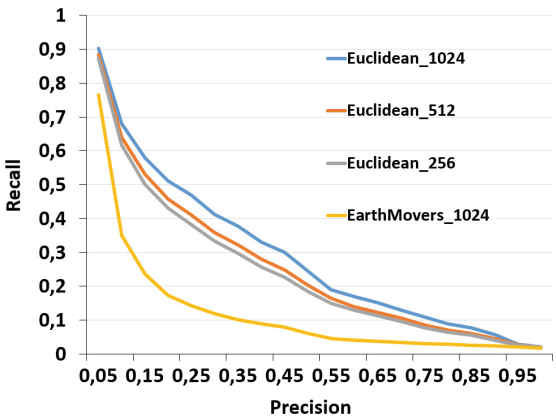

Fig. 3. Precision-recall curves obtained by our method with different settings.

Moreover, some other standard metrics [12] have been considered in our evaluation:

- Nearest Neighbor (NN): the percentage of objects belonging to the query class and ranked in the top k of the retrieval result where $k = 1$.
- First Tier (T1): the same idea as in NN where k depends on the size of the class query. If the class size is C then $k = C - 1$.
- Second Tier (T2): in this case $k = 2 * (C - 1)$.
- E-Measure (EM): the precision and recall calculated on the first 32 retrieved objects.
- Discounted Cumulative Gain (DCG): assuming that the user pays more attention on the first displayed results of a search, this measure assigns more weight to the relevant results located at the top of the list.

All these metrics are ranged in [0, 1] where 1 indicates the best performance. Using these metrics, we compared our best results (Euclidean distance calculated on 1024 dimensional vectors) with some of the most recent methods having exploited the SHREC2018 protein dataset. More precisely, we compared our

method (GLT4IP) with six methods described in [6]: 3D convolutional framework for protein shape retrieval (3D-FusionNet), Global Spectral Graph Wavelet framework (GSGW), Histograms of Area Projection Transform (HAPT), Protein Shape Retrieval driven by Digital Elevation Models (DEM), Scale-Invariant Wave Kernel Signature (SIWKS) and Wave Kernel Signature (WKS).

Table 1 summarizes the performances obtained by our method and by the six methods on the SHREC2018 protein dataset. It shows that our method GLT4IP reaches better results than GSGW, DEM and SIWKS. Three other methods outperform GLT4IP but this latter remains complementary since relatively fast outputs are obtained through the pre-trained CNN. Nevertheless, performances obtained by all current methods clearly show that characterizing the shapes of the proteins is not an obvious task, probably in reason of their high diversity and irregularity of shapes which make the current descriptors partially efficient.

Table 1. Performances of our proposed method GLT4IP compared to those of the state of the art methods obtained on the SHREC2018 protein dataset.

Method	NN	T1	T2	EM	DCG
GLT4IP	0.550	0.293	0.344	0.265	0.598
3D-FusionNet	0.689	0.404	0.459	0.366	0.681
GSGW	0.514	0.261	0.35	0.247	0.581
HAPT	0.77	0.493	0.584	0.462	0.755
DEM	0.421	0.238	0.319	0.231	0.555
SIWKS	0.199	0.109	0.189	0.114	0.452
WKS	0.717	0.41	0.49	0.377	0.701

4 Conclusion

The paper presents an approach (GLT4IP) indexing protein structures from associated 3D point clouds. The protein data is subsampled to fit with the input size of a CNN that was already pretrained onto man-made 3D object database. The subsampling stage is performed while keeping the shape topology. By subsampling data and transferring knowledge from a pretrained CNN, it makes GLT4IP relatively fast. GLT4IP performances overpass half of the state-of-the-art methods involved in the SHREC2018 contest. GLT4IP reveals the potential of a prepared transfer learning-based method for competing with research methods in protein shape retrieval.

Acknowledgments. The authors particularly thank F. Langenfeld, organizing member of the SCHREC 2018 challenge for his assistance and the double-check of the performance rates for our method presented in Table 1. They thank F. Malbranque, V. Tondeux, A. Jaffrezic and J. Xu for deploying the processing pipeline.

References

1. Benhabiles, H., Aubreton, O., Barki, H., Tabia, H.: Fast simplification with sharp feature preserving for 3D point clouds. In: 2013 11th International Symposium on Programming and Systems (ISPS), pp. 47–52, April 2013. https://doi.org/10.1109/ISPS.2013.6581492
2. Berman, H.M., et al.: The protein data bank. Nucleic Acids Res. **28**(1), 235–242 (2000). https://doi.org/10.1093/nar/28.1.235
3. Chandonia, J.M., Fox, N.K., Brenner, S.E.: SCOPe: manual curation and artifact removal in the structural classification of proteins extended database. J. Mol. Biol. **429**(3), 348–355 (2017). https://doi.org/10.1016/j.jmb.2016.11.023. Computation Resources for Molecular Biology
4. Hegyi, H., Gerstein, M.: The relationship between protein structure and function: a comprehensive survey with application to the yeast genome. J. Mol. Biol. **288**(1), 147–164 (1999). https://doi.org/10.1006/jmbi.1999.2661. Edited by G. von Heijne
5. Jin, X., Awale, M., Zasso, M., Kostro, D., Patiny, L., Reymond, J.L.: PDB-explorer: a web-based interactive map of the protein data bank in shape space. BMC Bioinform. **16**(1), 339 (2015). https://doi.org/10.1186/s12859-015-0776-9
6. Langenfeld, F., et al.: SHREC 2018 protein shape retrieval. In: Eurographics Workshop on 3D Object Retrieval, pp. 53–61, April 2018. https://doi.org/10.2312/3dor.20181053
7. Melo, R., et al.: A machine learning approach for hot-spot detection at protein-protein interfaces. Int. J. Mol. Sci. **17**(8) (2016). https://doi.org/10.3390/ijms17081215
8. 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 Computer Vision and Pattern Recognition (CVPR). IEEE (2017)
9. Ritter, J.: An efficient bounding sphere. In: Graphics Gems, pp. 301–303. Academic Press Professional Inc., San Diego (1990)
10. Rubner, Y., Tomasi, C., Guibas, L.J.: A metric for distributions with applications to image databases. In: IEEE International Conference on Computer Vision, pp. 59–66, Bombay, India, 9–13 May 1998
11. Rusu, R.B., Cousins, S.: 3D is here: point cloud library (PCL). In: IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011
12. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton shape benchmark. In: Proceedings Shape Modeling Applications, pp. 167–178 (2004). https://doi.org/10.1109/SMI.2004.1314504
13. Welzl, E.: Smallest enclosing disks (balls and ellipsoids). In: Maurer, H. (ed.) New Results and New Trends in Computer Science. LNCS, vol. 555, pp. 359–370. Springer, Heidelberg (1991). https://doi.org/10.1007/BFb0038202
14. Yang, H.: Protein surface analysis by dimension reduction with applications in functional annotation and drug target prediction. Ph.D. thesis, Drexel University (2015)