Protein–Protein Interface Topology as a Predictor of Secondary Structure and Molecular Function Using Convolutional Deep Learning

Benjamin Bouvier

Protein-protein interface topology as a predictor of secondary structure and molecular function using convolutional deep learning

Benjamin Bouvier*

Laboratoire de Glycochimie, des Antimicrobiens et des Agroressources, CNRS UMR7378/Université de Picardie Jules Verne, 10, rue Baudelocque, 80039 Amiens Cedex, France.

E-mail: benjamin.bouvier@u-picardie.fr

Abstract

To power the specific recognition and binding of protein partners into functional complexes, a wealth of information about the structure and function of the partners is necessarily encoded into the global shape of protein-protein interfaces and their local topological features. To identify whether this is the case, this study uses convolutional deep learning methods (typically leveraged for 2D image recognition) on 3D voxel representations of protein-protein interfaces colored by burial depth. A novel two-stage network, fed with voxelizations of each interface at two distinct resolutions, achieves balance between performance and computational cost. From the shape of the interfaces, the network tries to predict the presence of secondary structure motifs at the interface and the molecular function of the corresponding complex. Secondary structure and certain classes of function are found to be very well predicted, validating the hypothesis of interface shape as a conveyor of higher-level information. Interface patterns triggering the recognition of specific classes are also identified and described.

Introduction

Protein-protein (PP) complexes are the ubiquitous effectors of biological function. Recognition and interaction between protein partners occur along PP interfaces, which hold tremendous promise as druggable targets.[1] PP interface modulators can for example be used to correct the misregulation of interfaces involved in numerous diseases.[2–4] Within microbes, PP interface inhibitors can disrupt the formation of PP complexes implementing vital functions in a way that is much less prone to the outbreak of resistance than traditional active-site targeting drugs.[5–8]

Although their sizes and shapes can vary tremendously, PP interfaces are on average large and flat, with only a small proportion of interface aminoacids (termed hotspots) contributing significantly to the overall binding free energy.[9] Moreover, the interfaces of biologically relevant PP complexes may be hard to distinguish from those of transient complexes stemming from random interactions between noncognate partners due to crowding within the cytoplasm.[10] The putative relationship between topological and/or chemical features of PP interfaces on the one hand, and molecular function or biological process on the other, is thus very complex and has not been rationalized to date. In fact, choosing a self-contained set of minimally correlated features as a subspace in which to successfully categorize PP complexes remains an ongoing challenge.[11–13] Because of its synthetic and predictive power, deep learn-

ing is currently gaining traction for the study of PP interfaces based on sequence,[14–16] structural data,[17] or both.[18,19] However, the problem of selecting a feature space and efficiently encoding it for machine learning remains. A commonly used rationale to find trends in PP interfaces is to use a very large number of very diverse descriptors and let the learning algorithm pick the relevant ones. For example, Qiao et al use 82 distinct features, both local and global (one temperature, 10 physicochemical, 36 structural, 5 evolutionary, and 30 solvation properties).[20] While learning algorithms can detect simple correlations between features, their performance will strongly depend on the encoding of the features, the renormalization of their values to a common range, and the mix of global vs. local features, which are user-defined and far from trivial.

In this work, I use convolutional deep learning techniques to examine how much structural and functional information can be inferred from the global shape of a PP interface (defined as the surface equidistant to the atoms of both partners) and its accessibility to water. This is a crucial question: PP interfaces power PP recognition and thus probably encode, in a manner which is still unknown and probably quite complex, the high-level information required to specifically form a functioning complex. Deep learning techniques can identify and model complex nonlinear relationships within large datasets, but generally fail to provide an explicit representation of the latter and are thus often termed 'black boxes'. Convolutional networks differ in this respect: the patterns triggering recognition in the trained neurons can be extracted and visualized, often yielding valuable information on how the network 'sees' and categorizes the input data. This, in turn, should provide insights into the ability of interface topologies to power elaborate recognition schemes by acting as carriers for more complex structural or functional information.

Convolutional neural networks, which emulate the hierarchical detection of features by cells in the visual cortex, have proved very efficient in the field of 2D[21] and 3D[22] image recognition. Matrices of neurons called filters are arranged in consecutive connected layers; first-layer filters detect simple patterns (edges, textures, etc), activating second-layer filters which integrate multiple simple patterns into more complex ones, that are in turn fed to the next layer... This detection of patterns is robust to displacements, small deformations and noise. In recent years, convolutional networks have been increasingly employed for the prediction or classification of PP interfaces, based on 2D contact maps[18,19] or 3D images obtained by discretizing 3D structures into voxel sets (cubic volume elements akin to 2D pixels)[17,23,24]. In this work, the PP interface topologies, obtained as 3D polygon meshes using Voronoi diagrams of the partner atoms,[25] are discretized into 3D images using voxels which are colored according to their burial depth within the interface. Indeed, buried (desolvated) interface regions are known to be enriched in conserved hotspot residues, which are crucial to recognition and binding.[25,26] By using a global topological description colored by a single important feature relating structure to conservation, this study hopes to avoid the normalization pitfalls that come with using multiple features, staying close to the actual definition of a PP interface and leaving to the convolutional network the task of finding relevant local patterns from global data.

Thus, I build a dataset of more than 50000 PP interfaces voxelized at two distinct resolutions (coarse and fine). The dataset is used as input for a novel two-stage convolutional network which simultaneously takes information from both resolutions, capturing more surface detail than previous studies[17,23,24] while restraining computational cost. This novel dataset is used to tackle two learning problems. First, I look at whether interface topologies can be used to predict the presence of α-helix and/or β-sheet motifs at the interface. This is far from trivial: frequently occurring patterns of interacting secondary structure motifs have been identified,[27] yet the interface along which they form contains only a fraction of the atoms of such motifs, and their combination often results in flat patches with few salient features.[28] Is this

sufficient to make secondary structure a driving force of specific PP recognition? Predicting secondary structure from interface geometry is also quite important for *de novo* interface design[29,30] or the conception of PP interface modulators.[31] Second, I explore whether interface topology can predict the molecular function of a PP complex. This much more indirect relationship has not been explored to date; if it exists, it has profound implications on the amount of implicit information encoded within interfaces and would be a large step in understanding why random noncognate PP interactions do not give rise to stable complexes. Finally, considering that local secondary structure elements have been successfully linked to function,[32,33] relating both notions to interface topology appears a promising unifying goal.

Methods

Interface meshes

The dataset of PP complexes used in this study are the entries of the HIPPDB and SIPPDB databases of Arora and coworkers,[34] which categorize PP complexes in the protein databank (PDB) based on the nature of secondary structure motifs at their interface. The structures of all HIPPDB and SIPPDB entries were retrieved from the PDB. Missing or incomplete residues in the interface region were replaced, and their conformation optimized, using MODELLER.[35] Entries which contained only backbone atoms, or had been invalidated or superseded since the publication of HIPPDB and SIPPDB, were discarded. Water molecules were added at sterically available and energetically favorable positions within a 5 Å radius around the interface using SOLVATE;[36] this harmonizes PDB files with respect to the presence of structural water molecules.

The resulting structures were input into the Intervor module of the Structural Bioinformatics Library.[37] Intervor computes three binary interfaces AB, AW and BW between the two protein chains A and B and interfacial water molecules W, as the Voronoi power diagrams of the corresponding atoms. The water-mediated ternary interface ABW employed in this study was obtained as the union of these three binary interfaces. The resulting Voronoi facets were shelled from the rim to the core of the interface, associating to each facet an integer shelling order (SO).[25] SO represents the number of 'jumps' between adjacent facets needed to reach the interface rim from the current location, and thus is a good representation of burial depth; high-SO patches have been shown to correlate strongly with the presence of hotspots.[25] In PDB entries whith multiple interacting protein chains, the entire process was repeated for each of the pairs of chains forming an interface.

Particular dispositions of atoms at the interface rim can give rise to rim Voronoi facets with near-parallel edges which artificially extend very far from the interface. To prevent such unphysically large rim facets from raising issues during the voxelization process, the oriented bounding box of the un-rimmed interface mesh was computed, expanded by 10 Å in all dimensions, and used to clip the interface mesh. This 10 Å limit to rim facet lengths was chosen upon analysis of the distributions of rim facet lengths and areas, and resulted in 19 % of interfaces requiring clipping.

Voxelization

The voxelization and labeling processes described in the following paragraphs are summarized in a flowchart on Supporting Information figure S1.

Each interface mesh in the dataset was voxelized at two resolutions: 1 Å (fine) and 4 Å (coarse). This was done by shooting rays along the positive z direction from a regular grid of points in the xy plane, converting the intersection points of these rays with the interface mesh to voxel locations, and coloring the latter with the SO of the intersected facets (or zero when no intersection occurs). The fine and coarse voxel grids were cubes measuring 128 and 32 voxels to a side, respectively, spanning 128^3Å^3. This represents a much higher resolution than previous studies using voxel representations of proteins or interfaces.[17,23,24] 423

complexes (0.7 % of the dataset) featuring interface meshes too large along at least one dimension to fit within the grids were removed from the dataset. The dataset thus obtained consisted of 56864 interfaces, each associated with a coarse and a fine voxel representation, 3D images that can directly be fed into convolutional layers. Figure 1 demonstrates the voxelization of a sample interface.

Labeling the dataset

Depending on the learning task, the dataset PP interfaces were labeled based on either their secondary structure motifs or the molecular function of the PP complexes.

For learning interfacial secondary structure motifs, the labels α and β were attributed based on the information in the HIPPDB and SIPPDB databases. The labels were one-hot encoded within a 2-vector (one vector component per label, holding 1 if the label is set or 0 otherwise; entries containing both α and β motifs have both labels set simultaneously). All 56864 entries were thus labeled (35206 α, 13867 β, 7791 both).

For the optimal learning of molecular function, labeling the largest part of the dataset PP interfaces with the smallest number of function-related tags is required. Gene Ontology terms (GOTs),[38] which annotate biological systems in terms of molecular function, cellular component and biological process, appeared as a promising source of labels. The 'slim' subset of GOTs, curated by the the Gene Ontology Consortium, was employed; it provides a broad overview of functions, locations and roles by hierarchically grouping multiple related GOTs.[39] The molecular function GOTs associated with each partner of the dataset complexes were retrieved using PyPDB.[40] 15286 dataset entries lacked GOT annotations altogether and 8814 more had no molecular function-related GOT. The remaining 32764 entries were described by 37 unique GOTs, with tremendously varying representativity (from 30 instances of 'histone binding' to 20380 instances of 'ion binding' PP complexes – see Supporting Information figure S2).

This is not ideal: to facilitate machine learning, class labels should verify the following criteria. First, they should be available for the largest possible proportion of the dataset. Second, the number of dataset entries per label should be as balanced as possible. Third, the labels should be as independent from each other as possible: their pairwise semantic similarity[41] should be as small as possible, and the average number of labels per entry should be as close to one as possible. Simultaneoulsy optimizing these criteria, however, proved impossible. For instance, maximizing the number of retained dataset entries led to the selection of a few highly represented GOTs, resulting in large discrepancies in the number of entries described by each GOT; conversely, maximizing the homogeneity of the number of entries labeled with each GOT led to selecting GOTs that are not as highly represented, drastically reducing the number of labeled dataset entries. This is pictured on Supporting Information figure S3, which shows the pairwise correlation between the criteria, while tables S1-S4 present the subsets of GOTs optimizing each criterion.

To identify the set of GOTs representing the best tradeoff, a score function was built upon the normalized contributions of the above criteria and all possible GOT sets sufficiently small to allow efficient multilabel deep learning[42] were scored (see Supporting Information figure S4). The top-scoring set of labels, of length 10, was adopted as a descriptor of molecular function for the PP complexes in the dataset. It is presented in table 1. 31139 entries were successfully labeled with it, a significant decrease from the original dataset size but sufficient to train a 10-label deep classifier network. Due to the scarcity of entries having both verified structures and function, the initial dataset is already unbalanced in terms of protein families or function and the culling should have no major detrimental statistical effect. Additionally, the average pairwise RaptorX TM-score[43] of structures within each class is within the range observed for unrelated proteins, which shows that the classes are not dominated by a small number of folds (see Supporting Information figure S9. As previously, one-hot encoding was employed to represent each entry's labels as a

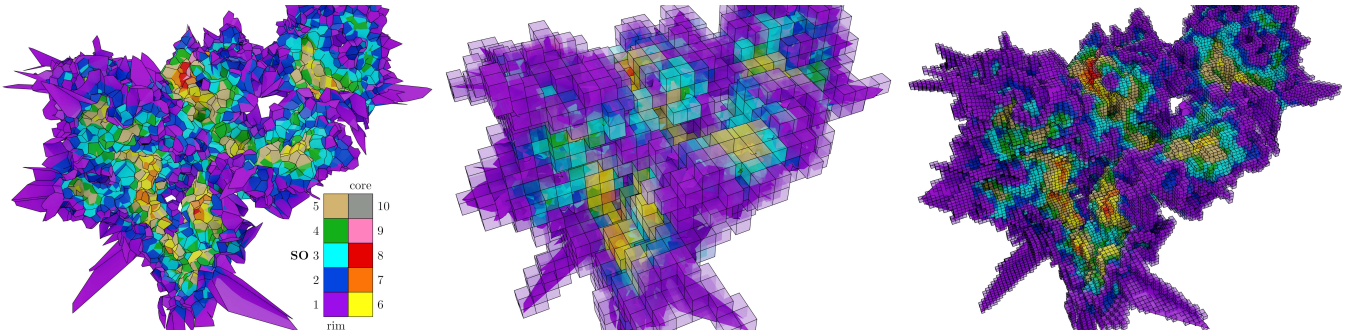

Figure 1: Multi-resolution voxelization of a sample PP interface. Left: Voronoi interface. Center: coarse voxelization (4 Å resolution). Right: fine voxelization (1 Å resolution). Voronoi facets and voxels are colored by SO value; the color scale will be used in all figures of the manuscript.

Table 1: Selected set of 10 GOTs used to describe molecular function. This set is a tradeoff between maximizing the number of dataset entries, minimizing semantic similarity between GOTs, minimizing the number of GOTs per entry, and minimizing the standard deviation of GOT populations over the dataset.

GOT id	GOT	Population
GO:0016491	oxidoreductase activity	11224
GO:0008233	peptidase activity	5854
GO:0016829	lyase activity	3878
GO:0003677	DNA binding	2901
GO:0022857	transmembrane transporter activity	2864
GO:0016301	kinase activity	1691
GO:0016853	isomerase activity	1622
GO:0005198	structural molecule activity	1309
GO:0016810	hydrolase activity, acting on carbon-nitrogen (but not peptide) bonds	1228
GO:0016874	ligase activity	1084

10-vector of zeros and ones.

Network topology

The computational cost of 3D convolutional networks increases rapidly with the resolution of the filters. To alleviate this effort while preserving the ability to learn from high-resolution interface features, this work employs a two-stage convolutional neural network schematized on figure 2. The main stage (dubbed CLC) is a label classifier which takes as input the coarse ($32 \times 32 \times 32$) voxelization of a PP interface and outputs an n-vector of probabilities that the input carries each of the n labels. CLC is composed of a succession of three convolution-maxpooling-ReLu blocks of decreasing size and increasing depth, a dropout layer, and a series of densely connected layers, with the final layer's depth equal to the number of labels. The convolution layers have a kernel size of $3 \times 3 \times 3$, a stride of 1 and a padding of 1; the maxpooling layers have a kernel size of $2 \times 2 \times 2$, a stride of 2 and no padding; such relatively small, overlapping kernels were found to give the best results, as they also do for 2D image classification.

An auxiliary stage of similar composition (denoted F2C) predicts the scalar value of a coarse voxel from the corresponding $4 \times 4 \times 4$ fine voxels. F2C basically performs a dimensionality reduction operation, choosing the most salient high-resolution features to encode into the low-resolution representation; it thus yields potentially more relevant low-resolution voxelizations than the simple ray-casting procedure described above. In each forward pass of the two-stage network, a user-defined proportion of randomly selected coarse voxels are replaced by the cor-

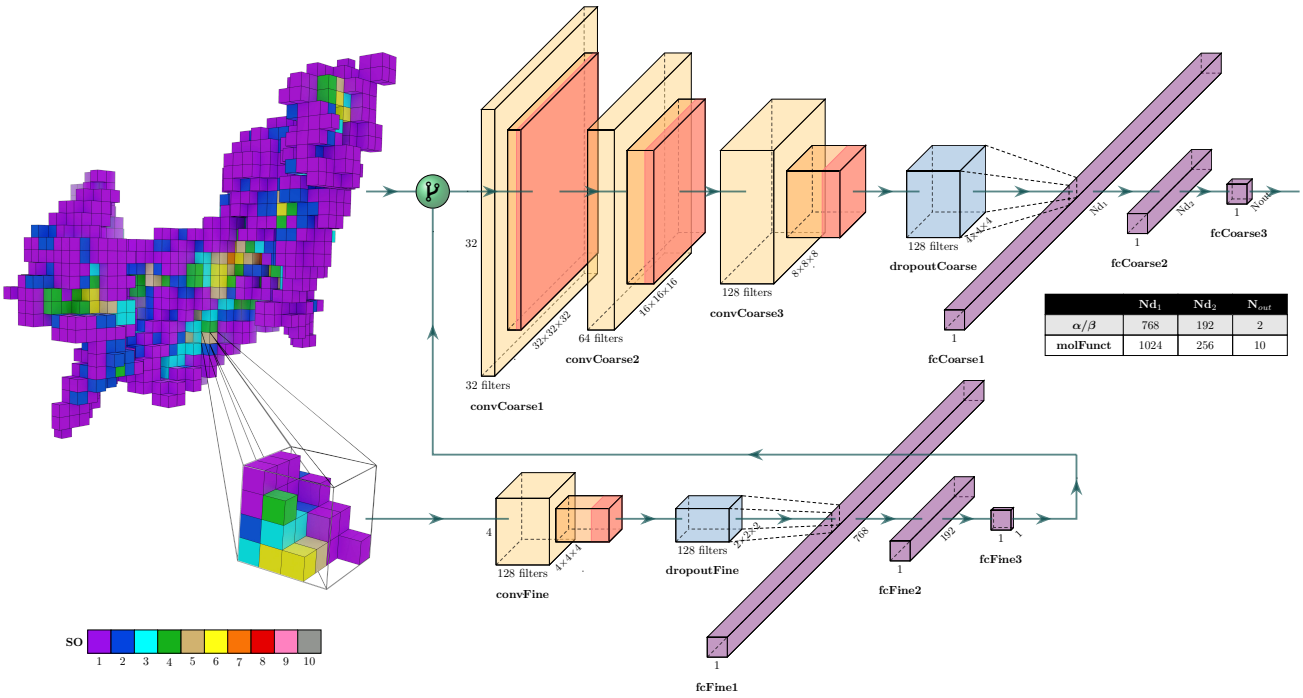

Figure 2: Architecture of the two-stage CLC/F2C convolutional network. The nature of a layer is indicated by its prefix (conv=3D convolution, fc=fully connected). Left: $32 \times 32 \times 32$ *(coarse) voxelization of an example interface, colored by shelling order (SO), and the decomposition of a coarse voxel into its* $4 \times 4 \times 4$ *subset of fine voxels, both of which are used as input to the neural network. The depths of the fully connected layers of the CLC subnetwork, which depend on the number of labels to predict, are given in a table (α/β: interfacial secondary structure elements; molFunc: molecular function).*

responding predictions of the F2C stage before the coarse voxelization is input into CLC. A value of 40% was found to be an acceptable tradeoff between accuracy and computational cost.

Deep learning

The dataset was split into a training and a test set of respective populations 75% and 25%. CLC was pretrained to predict the correct label vector from the coarse voxelizations of the training set PP interfaces. F2C was pretrained on $4 \times 4 \times 4$ voxel blocks randomly chosen from the fine voxelizations of the training set interfaces using an autoencoder, obtained by plugging a mirror image of F2C into the output of F2C and predicting the input voxel blocks from themselves. In such an autoencoder, the input data fed into the network passes through a bottleneck, at which point the network must choose what information to keep and what to discard so that the input data can best be reconstructed.[44] Once trained in this fashion, the F2C subnetwork is thus guaranteed to provide coarse voxel values that encapsulate the most relevant information contained in the corresponding fine voxels.

CLC and F2C were then combined as per figure 2, and their pretraining was refined to predict labels from the combined coarse and fine voxelizations of the training dataset. Binary cross-entropy with logits was used as the loss function, and the network weights were optimized using stochastic gradient descent. To reduce overfitting, dataset augmentation was performed by applying to the training set samples, with a probablity of 15 %, a random number of $\pi/2$ rotations along each of the three base axes **x**, **y** and **z** and a random number of flips along the three base planes **xy**, **xz** and **yz**. All deep learning tasks were implemented using the PyTorch API.[45]

Clustering of voxel motifs

The clustering of voxel motifs maximizing the activation of individual convolution filters was performed as follows. In each voxel set, a blob detection algorithm[46] was employed to detect contiguous zones of high SO. A graph was then built, using the blob diameters as node weights and the distance between blobs as edge weights. The set of graphs associated with all convolution filters were then clustered by means of the spectral clustering method,[47] using graph edit distance[48] as a metric to compare graphs with one another. The number of clusters was chosen to maximize the silhouette score and minimize the Davies-Bouldin score, both well-documented descriptors of clustering performance.[47] To obtain a finer clustering, a hierarchical approach was employed: the clusters were recursively subdivided into subclusters as long as no major degradation of the silhouette or Davies-Bouldin criteria was observed and the subclusters remained relatively balanced in size (i.e., no singleton clusters). According to these criteria, a clustering depth of 2 was found to be optimal for both learning tasks considered. To facilitate the viewing as 3D images of the representative voxel motifs obtained from the clustering, they were denoised using a wavelet filter.[46]

Results

Prediction of α/β interface motifs and PP complex molecular function

Instances of the two-stage network on figure 2 were trained (on the training set) to predict the presence of α or β motifs at the interface and the molecular function of the PP complex. The learning performance was subsequently evaluated (over the test set) by measuring the per-class prediction accuracies and F1 scores. As a benchmark of the dual-resolution methodology, the benefit of using the F2C stage was evaluated by comparing the predictive performance of the full network with that achieved by the CLC stage only.

Accuracy is a straightforward, easy to interpret measure which is proportional to the sum of true negative and true positive predictions; however, in datasets with multiple unbalanced classes (like molecular function in the present study), underrepresented classes have very high true negative scores by definition and feature artificially high accuracies even if the network is unable to predict them (low true positive score). Supporting Information figure S5, which shows the prediction accuracies for the 10-class molecular function problem, illustrates this clearly. In such cases, the F1 score, a mixture of precision and recall which does not involve true negatives, is considerably more informative of the prediction quality. However, the F1 score is much less intuitive than accuracy and is also affected by class imbalance. To alleviate this, the normalized score $\mathrm{F1}_{\mathrm{norm}}$ is used here: $F1_{norm} = (F1 - F1_{rand})/(1 - F1_{rand})$, where $\mathrm{F1}_{\mathrm{rand}}$ is the F1 score achieved by a random classifier for predicting the selected class over the considered dataset. $\mathrm{F1}_{\mathrm{norm}}$ thus ranges from 0 (random predictor) to 1 (perfect predictor).

The converged accuracies and normalized F1 scores for the prediction of secondary structure are shown in table 2; for molecular function, only $\mathrm{F1}_{\mathrm{norm}}$ is shown since accuracies are misleading (table 3). The evolution of these statistics with the number of training epochs can be found on Supporting Information figures S6 and S7.

Table 2: Accuracy and $F1_{norm}$ values for the prediction of α and β motifs using the CLC subnetwork only or the complete F2C+CLC network.

	CLC		CLC+F2C	
	Acc.	$\mathrm{F1}_{\mathrm{norm}}$	Acc.	$\mathrm{F1}_{\mathrm{norm}}$
α	0.94	0.66	0.94	0.86
β	0.90	0.73	0.91	0.85

As can be seen, excellent accuracies and F1 scores are achieved for the prediction of α and β motifs at the interface using the complete CLC+F2C network. The F2C subnetwork does not significantly improve accuracy, but has a

Table 3: $F1_{norm}$ values for the prediction of molecular function using the CLC subnetwork only or the complete F2C+CLC network.

Function	CLC	CLC+F2C
oxidoreductase	0.63	0.71
peptidase	0.79	0.82
lyase	0.33	0.40
DNA binding	0.43	0.48
transporter	0.67	0.73
kinase	0.26	0.33
isomerase	0.30	0.36
structural	0.47	0.56
hydrolase	0.25	0.35
ligase	0.38	0.45

marked effect on the F1 score, which is actually more relevant since the dataset is somewhat unbalanced (75.6 % α vs 38.1 % β motifs). From these results, it can unambiguously be claimed that the overall interface shape encodes, and is specific of, interfacial secondary structure motifs.

In the case molecular function, the network achieves much better scores than a random classifier for the prediction of all functional classes but overperforms for oxidoreductase, peptidase and membrane transporter activities compared to kinase or hydrolase. Interestingly, the scores obtained are higher than those achieved using other structure-based predictors, despite the much larger number of features included in the latter.[49,50] They are similar on average to those obtained by Amidi et al[24] on voxelized representations of protein backbones (once the F1 score provided by these authors has been normalized). Both approaches perform better for oxidoreductases (0.74 vs 0.71) and worse for ligases (0.49 vs 0.45); interestingly, the Amidi approach predicts hydrolases rather well (0.74), whereas the present method differentiates peptidases which are very well predicted (0.82) from other hydrolases for which prediction is difficult (0.35).

The F2C subnetwork substantially improves the prediction of all classes; as can be seen on Supporting Information figure S7, it also prevents the onset of overfitting which tends to occur beyond 100 training epochs for the CLC network, causing the F1 curve to plateau and decrease whereas the CLC+F2C score continues to slightly rise. Encouragingly, the prediction performances seem decorrelated from the label populations, making it likely that the observed trends are due to the interface topologies themselves and their diversity within each functional class rather than dictated by statistical artifacts. Finally, adding convolutional or dense layers or increasing their size did not result in a significant boost to F1 scores (data not shown), which implies that the limiting factor is probably the quality of the dataset, the voxel resolution, or the actual information contained in the interfaces.

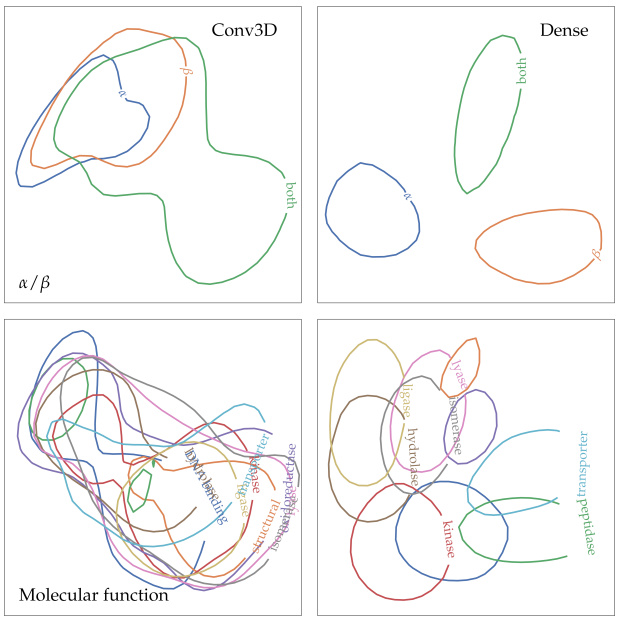

Figure 3: Density contours of the activations of the 3D convolution filters (left column) and fully connected neurons (right column) for samples tagged with a given label (top row: secondary structure; bottom row: molecular function), projected on the first two principal components of the activations of the complete dataset.

Analyzing layer activations

I now examine the activations of the 224 individual 3D convolution filters of the trained networks (respectively 32, 64 and 128 in layers 1, 2 and 3) when instances of interfaces bearing α or β motifs, or belonging to each of the 10 molecular function classes, are presented to them. The

activations of the neurons in the fully connected (dense) layers (962 for the prediction of secondary structure and 1290 for molecular function) were similarly studied. Principal component analyses (PCA) were performed over the set of vectors containing the activations of all dataset samples. These activations were then projected onto the two first eigenvectors (which explained more than 90 % of the total variance over the dataset for both learning tasks and both dense and convolutional neurons). The density plots of these projections are depicted on figure 3. As can be seen, the activation patterns for the convolutional layers do not seem to clearly distinguish labels from one another, whether for the prediction of secondary structure or molecular function: the corresponding density plots overlap each other to a large extent. Interestingly, the point cloud for interfaces bearing both α and β motifs extends into a region of eigenvector space distinct from that corresponding to α or β only, hinting at the existence of specific patterns for such interfaces. On the contrary, the activation patterns of the dense neurons clearly distinguish classes from one another. This is most apparent for the prediction of α and β motifs; once again, PP interfaces simultaneously bearing both motifs occupy a distinct region spanned by the second PCA eigenvector. However, to a large extent, the dense layer activations also manage to disentangle the 10 different classes of molecular function. This means that the interface patterns detected by the convolutional filters are generally not specific to a given interface class: instead, the actual prediction is performed downstream by the fully connected layers which aggregate and synthetize the information about the motifs detected by the convolutional layers, achieving specificity.

Visualizing convolution filters

To visualize what types of interface patterns are recognized by the network, the input voxels that maximize the activation of a given convolution filter can be generated using a technique inspired from neural style transfer.[51,52] Starting from a random set of input voxels (e.g. Perlin noise), a forward pass through the network is performed; the average activation of the filter and its gradient relative to the input are computed. The gradient is then used to iteratively update the input voxels in a way which specifically maximizes the activation of the chosen filter. The voxel motifs maximizing each of the 224 convolution filters in the three layers were thus computed.

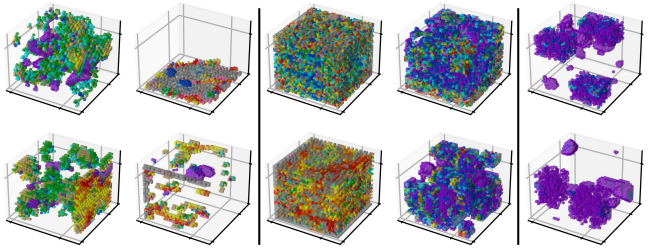

Figure 4: Typical examples of voxel patterns maximizing the activation of convolution filters in the network's first (left), second (center), and third (right) convolution layers. Voxels are colored by SO value.

As is typically observed in neural style transfer methods applied to 2D images, the obtained voxel motifs feature complex patterns at different resolutions. Relevant exemplars are rendered on figure 4. Patterns activating first-layer neurons look the most like actual interfaces, with features that are easily distinguishable from background noise. These can consist of zones of high SO surrounded by layers of progressively decreasing SO, which are either sets of localized 'hotspots' or axis-oriented planes or lines; the combination of several such elementary patterns is likely to match most of the real-life interfaces encountered in the dataset. Patterns detected by filters in the second layer tend to be larger and denser. They either consist of high-frequency, noisy voxel distributions covering the entire filter, spanning a limited range of SO values (whether high or low) and bearing little resemblance to actual interfaces; or they appear as large blobs separated by blank/rim voxels, with a progression of SO values that are much closer to real interfaces. Finally, third-layer filters appear much sparser, consisting of isolated, often parallelepipedic blocks of low-SO voxels surrounded by rim voxels. Encouragingly, the motifs maximizing filters from all three layers were found to be roughly indepen-

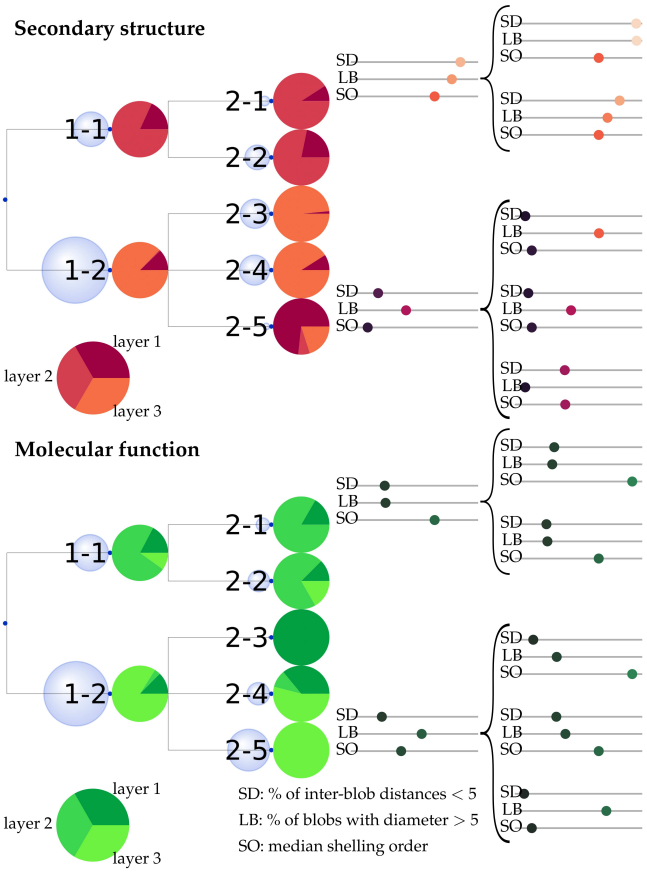

Figure 5: Hierarchical clustering of voxel motifs maximizing the activation of the 3D convolutional filters. Top: prediction of secondary structure motifs; bottom: prediction of molecular function. For each cluster, the distribution of filters within the three layers of the network is shown as a pie chart. The average proportion of small inter-blob distances (SD), the average proportion of large blobs (LB) and the median shelling order (SO) are also shown as dots on a horizontal scale. The population of each cluster is proportional to the diameter of the blue circle next to to its name.

dent of the Perlin noise pattern used to initialize the optimization process, with most apparent differences disappearing in less than 10 iterations. Clearly, although the activation functions are highly complex and nonconvex, their local maxima all share a common nature and the voxel motifs presented herein are representative.

To obtain a synthetic view of the different types of voxel motifs maximimizing filter activations, the motifs were hierarchically clusterized as described in the Methods section. Both for secondary structure and molecular function predictors, this resulted in 2 first-level clusters (denoted 1-1 and 1-2), further subdivided into 5 second-level clusters (labeled 2-1 to 2-5). They are presented on figure 5. The voxel motifs contained within each cluster were analyzed based on the distribution of high-SO voxel groups or 'blobs' (distance between blobs, typical blob size and SO), as well as on the layer containing the convolutional filter which each voxel set maximizes. Unsurprisingly, both for the prediction of α/β elements and for that of molecular function, the first level of clustering clearly separates second- (cluster 1-1) from third-layer filters (cluster 1-2): the former are dense, noisy and target high SO, while the latter are sparse and feature low, homogeneous SO values. First-layer motifs, very diverse in terms of density, are distributed in both clusters.

For the prediction of α/β elements, the second level of clustering splits the dense and noisy motifs of cluster 1-1 into two subclusters, regrouping in 2-1 most of the first-layer motifs along with the less dense and more interface-like exemplars of second-layer motifs. Cluster 1-2 is split mainly based on typical blob size: 2-3 and 2-4 are built from third-layer filters and feature isolated small (2-4) or medium-sized (2-3) blobs of low SO, surrounded by rim voxels; 2-4 also contains first-level filters which contain axis-aligned 2D or 1D patterns. Finally, 2-5 isolates a small number of sparse first-layer filters detecting localized SO hotspots.

Molecular function filters follow a similar clustering trend. At the first level, the segregation between second- and third-layer filters (in 1-1 and 1-2, respectively) is still apparent but not as perfect as for the prediction of secondary structure. At the second level, 2-1 and 2-2 differ mostly by SO values, with 2-2 regrouping third-layer and lower-SO second-layer filters. 2-5 isolates sparse, low-SO third-layer motifs from denser motifs in 2-4, including most first-layer ones. Finally, 2-3 regroups a few first-layer motifs targeting sparse high-SO hotspots. On the whole, compared to the α/β case, blobs are smaller and inter-blob distances larger, and first-layer filters tend to react to higher SO values.

Having regrouped the patterns activating convolutional filters into clusters, it is now much simpler to examine how the different in-

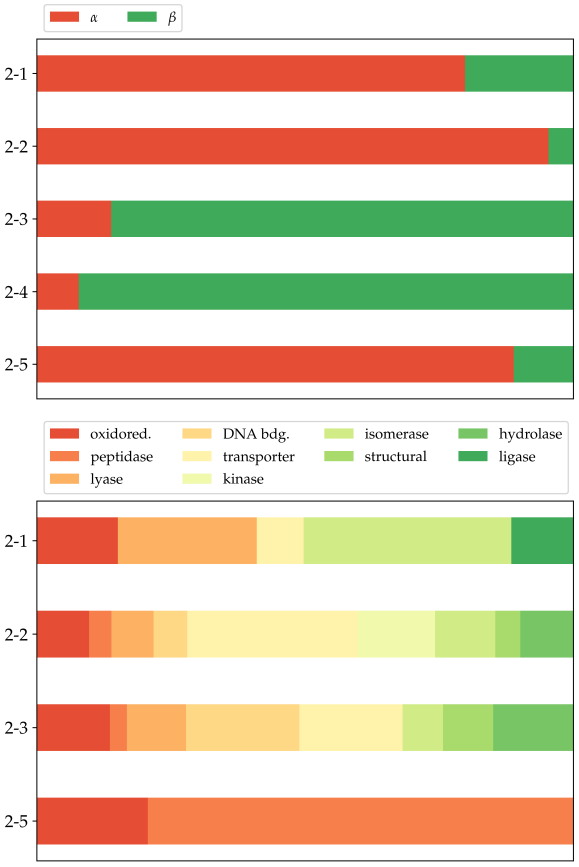

Figure 6: Label distributions of the 50 dataset entries that maximize the activations of convolutional filters inside a given cluster, for the prediction of secondary structure (top) and molecular function (bottom). The distributions have been corrected to account for the relative population of each label in the entire dataset.

terface types in the dataset activate the network. For each second-level cluster, the 50 PP interfaces which predominantly activate this cluster over all other clusters were identified, and statistics were performed over their labels (corrected by the relative populations of labels in the entire dataset). The results are shown on figure 6. For the prediction of secondary structure, they are striking: clusters 2-1, 2-2 and 2-5 (built mainly on second-layer filters) respond overwhelmingly to α patterns, while 2-3 and 2-4 (favoring third-layer filters) appear specific to β patterns. This means that secondary structure is mainly detected in the second layer of the network, which combines together patterns detected in the first layer. The third layer effectively performs a logical NOT operation on the second-layer results and thus appears largely redundant: predicting secondary function seems to be a relatively simple task that does not require very deep networks. Another important finding is that the specific response of clusters to α or β motifs is not only due to the second- and third-layer motifs which make up most of the clusters' populations. Indeed, although first-level filters in 2-1 and 2-2 are always more activated than that in 2-3 and 2-4, the activation of the latter is 2.31 times superior in the case of β instances compared to α (data not shown). These sparse, low-SO filters thus act as a correction to the baseline provided by the dense, high-SO filters in 2-1 and 2-2, switching recognition from α to β.

Figure 7 shows samples from the 5 top-scoring interfaces for the activation of each cluster. Clusters 2-1 and 2-2 favor highly curved interfaces that have large, continuous cores with high SO; all of the top 5 activators of 2-2 are relatively small, wedge-shaped hydrophobic 'pockets' with a small rim/core ratio, while those of 2-1 are larger and have extensive rims. Top triggerers of clusters 2-3 and 2-4 tend to feature lower SO; in 2-3, interfaces consisting of several, sometimes disconnected hotspots appear prominently, while 2-4 mainly has small interfaces among its top 5. Finally, interfaces maximizing 2-5 are more diverse but mostly look isotropic and relatively flat. Overall, β activation patterns are compatible with interfaces that are smaller and more accessible to water than their α counterparts, which could be linked to the typically larger solvent exposure of the β-sheet backbone.[53]

Conversely, the correlation between molecular function and activated filter cluster (figure 6) is not as marked. The activation of cluster 2-4 was not found to be dominant in any PP interface. Cluster 2-5, which contains most third-layer filters, reacts strongly to peptidase activity, which it distinguishes from oxidoreductase activity. Since these two are the most represented classes in the dataset, it makes sense that the network would allocate a large number of filters to classify them, but it is interesting that these belong to the third layer, which synthesizes outputs from both previous layers into large-scale patterns. On the contrary, clusters 2-1 to 2-

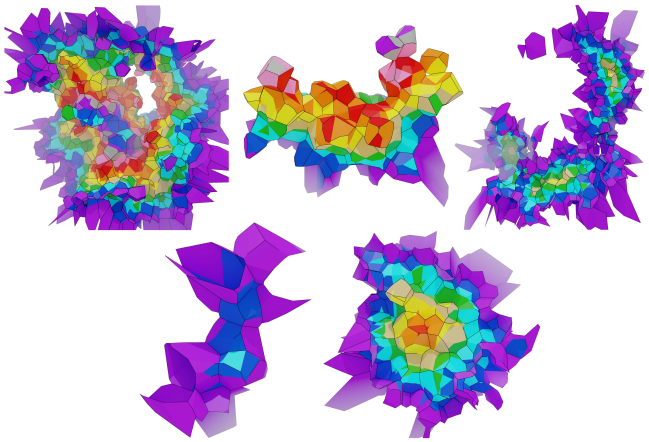

Figure 7: Examples of Voronoi interfaces maximizing cluster activations for the prediction of interface secondary structure. Top row, left to right: cluster 2-1 (PDB Id. 1P5R), 2-2 (2VLL), 2-3 (1EZV). Bottom row, left to right: 2-4 (1AVO), 2-5 (4GBI).

3 which react to most functional classes except peptidase mainly encompass second-layer filters which operate on a more local scope. Peptidase thus appears to be an outlier, whose detection requires a more global view of the interface topology than any other functional class. Other examples of relative specificity in the activation of filter clusters include ligases and isomerases, which predominantly trigger cluster 2-1. The filters detecting non-enzymatic classes (structural molecules, transporters, DNA binding) tend to be distributed between clusters 2-1, 2-2 and 2-3. On the whole, like for the prediction of secondary structure, classification seems to be mostly performed in the two last layers from common elements detected by the first layer; however, for this more complex prediction task, the last layer neurons are used to their full extent.

Discussion

Interfaces power the intricate mechanisms of PP recognition; therefore, they necessarily convey a wealth of information which it is essential for a successful machine learning encoding to preserve. However, maximizing the number of descriptors in an attempt to capture this information as completely as possible can be counterproductive: the bias introduced by the relative weight of each descriptor, whether explicitly set or inherent to the encoding, is difficult to evaluate and control. The aim of this study is to use a simple yet robust representation of global interface shape, implicitly taking sequence effects into account via the correlation between residue hydrophilicity/hydrophobicity and water accessibility/burial depth. Convolutional networks are then leveraged to extract salient local features from this global representation, something they have proven to excel at doing in the field of image recognition. This approach solves in an elegant way, and with minimal human intervention, the difficult problem of achieving a balanced mix of local and global features to use in a successful predictor of structure and/or function. Popular measures of structural similarity (Dali,[54] TM-align...[55]) tend to prioritize global shape (fold), which is often successful at predicting function; yet the functions of structurally similar proteins can be very diverse and variations upon a conserved functional core can lead to different folds,[33] making the consideration of local features indispensable.[32] The PRISM method,[56] for instance, is quite successful at predicting function from small sets of secondary structure motifs. This work follows this logic and takes it one step further by hypothesizing that if local secondary structure motifs can be predicted from a global representation of the interface, so can function. On the whole, the hypothesis appears verified. In fact, the main caveat probably does not lie with the method itself but with finding functional labels of balanced populations for optimal deep learning (since only a minute fraction of entries in sequence databases have both a structure and a verified function,[57] this issue is likely to endure). The method should prove a valuable addition to existing de novo interface design tools,[58] which are used to suggest partners that bind to a target protein along an interface of known shape.

Prediction of α and β interface motifs from interface topology is excellent; the near-redundancy of the third convolutional layer shows that the network is well-dimensioned to deal with the problem. The correlation between secondary structure and interface shape thus appears relatively straightforward, which is far

from trivial considering that in 60 % of interface helices, only one residue out of three actually has atoms at the interface.[28] The slightly better prediction of α over β motifs could be due to the fact that the former tend to be better conserved[59] and conservation is strongly correlated with SO[25,26] which is used to color the voxels. Helical motifs are also known to allow the binding of different partners to a single site: helices are robust to changes in side-chain identities and variations in local packing, allowing alternate ways to achieve binding.[60] This variability in helical patterns probably adds noise to the interface topology dataset, which is known to facilitate deep learning by preventing overfitting and facilitating generalization. The fact that PP interfaces combining α and β motifs activate specific recognition patterns distinct from those of either motifs is quite intriguing, and suggests the existence of collective effects due to specific arrangements of local motifs. Mixed α-β patterns have indeed shown remarkable properties in chimeric oligomers, for instance as rigidified α helix mimics.[61]

Protein domains are known to act as functional units. The ProtCID database,[27] which clusters structurally similar interacting motifs within PP complexes featuring identical Pfam domains,[62] hints at the existence of a link between structure and function at the domain level. Indeed, although Pfams tend to be defined based on sequence and structure, their correlation with GOTs is now established.[63] The present study confirms this: more than 90 % of the members of the ProtCID clusters containing the top activators of the convolutional filter clusters (figure 6) were found to share at least one molecular function GOT (see Supporting Information figure S8 for details). The correlation with Pfams also reinforces the relevance of GOTs as descriptors of function in this study; compared to Pfam clans, the more detailed hierarchical relationship between GOTs facilitates the selection of functional label subsets for the generation of balanced datasets.

Despite this, the prediction of molecular function from interface topology does not appear as staightforward and heavily depends upon the functional class. This is not entirely surprising considering that only a fraction of the aminoacids of interacting Pfam domains actually contribute to the interface topology. A similar trend has been observed by other researchers: by clustering graphs of residues, Saha et al showed that while larger domains are rather specific of a given enzyme class (with the size and diversity of such specific domains depending on the class), frequently occurring small structural motifs at the interface are common to all six classes (oxidoreductase, transferase, hydrolase, lyase, isomerase, ligase).[64] The link between interface topology and function is thus understandably indirect. Interestingly, however, the prediction of function from interface topology studied herein performs no worse than structural methods based on entire protein structures.[24] This fortifies the idea of conserved functional cores[33] of intermediate sizes indirectly impacting interface topology as well as overall structure.

Also noteworthy is the fact that the prediction of function by the network goes beyond the simple recognition of proteins sharing a similar fold. As shown in Supporting information figure S9, the average pairwise RaptorX TM-score[43] among the main contributors to the activation of convolution filter clusters is lower than 0.4, which corresponds to a 90% chance of having different folds. Understandably, the relationship to the fold is more marked for the prediction of secondary structure elements, but the corresponding TM-scores are still remarkably low. This shows the ability of convolutional deep learning to detect finer trends at different scales.

Saha et al[64] show that motifs found in hydrolases have the lowest overlap to motifs of other classes; my results hint that this is mostly due to peptidases (which are detected by the network with excellent specificity) rather than to other hydrolases. Interestingy, Saha et al find no overlap between oxidoreductase and hydrolase motifs (even smaller ones), yet in the present work these two classes tend to activate the same convolutional filters which mostly belong to the third layer. This means that the overall disposition of motifs on a

global scale (detected by the third layer) is as important as the motifs themselves, a testimony to the importance of mixing global and local effects which the F2C+CLC network strives to achieve. Results on the prediction of nonenzymatic complexes are also quite interesting. While it is not surprising that structural molecules, which can be quite diverse, are difficult to predict, the good score achieved by the network on membrane transporters is intriguing. Indeed, sequence similarities between transporters, whether within the same substrate transport subclass or for transporters of different substrates, is usually very low.[65] However, it has been shown that weakly stable regions in the transmembrane domain of transporters are often implicated as PP interfaces, with relatively little conformational entropy variation upon binding;[66] such conformational freedom could introduce noise in the corresponding interface topologies, boosting the learning process.

Both the prediction of secondary structure and molecular function clearly benefit from the F2C subnetwork. While it is easy to conceptualize that incorporating information from fine voxelizations into coarse models of the PP interfaces is beneficial, another more subtle consequence exists: by modifying the value of random coarse voxels, the F2C subnetwork also helps to regularize the dataset, preventing overfitting (akin to the effect of data augmentation or a dropout operation). Interestingly, considering that regularization is often performed by adding random noise to the dataset, this added benefit of F2C is expected to be quite independent of the actual performance of the subnetwork, facilitating the learning process even for interfaces that are not well dscribed by F2C's dimensionality reduction scheme. In addition, the independence of the F2C and CLC subnetworks enables them to be trained independently from one another on the same dataset, and the weights transfered to the complete network whose training then only needs to be refined. This type of transfer learning has been proved to provide excellent results at a low computational cost.[67]

Finally, the activation patterns of the convolution filters are not as easy to interpret as those observed on 2D image sets. As for the latter, first-layer filters consist of a variety of simple, localized features. The occurrence of axis-oriented lines and planes is quite interesting: it could represent the network's response to the fact that the input data is inherently 2-dimensional, composed of 2D facets separated by 1D edges, and is reminiscent of Bau et al's finding that the directions of the basis vectors are more meaningful than random directions in 2D convolutional network activations.[68] Also interesting is the prevalence of high SO values among patterns activating first-layer filters; this is in line with the long-standing theory that deeply buried, hydrophobic aminoacid hotspots dictate PP recognition and binding.[69] Downstream convolutional layers aggregate the information of upstream layers via maxpooling; in 2D images, this generally translates into larger and more complex activation patterns when moving toward deeper convolutional layers. Here, on the other hand, many second-layer motifs appear quite noisy. Although high-frequency noise patterns are inherent to strided convolution and pooling operations,[70,71] in this case they can be particularly difficult to separate from the signal. Nevertheless, large patterns looking like actual interfaces do occur for many filters of the second layer and the majority of the third. In these, rim voxels (SO= 1) play an important role: unlike hotspots which are localized, solvation effects require a more global view of the PP interface. This is yet another manifestation of the combination of local and global features inherent to the method.

Despite its successes, the method has room for improvement. For starters, encoding a 2D surface using 3D voxels is rather inefficient. Typical datasets of 2D or 3D images used for convolutional deep learning usually do not have lower effective dimensionalities. Scaling is also much less favorable in three dimensions than in two: the size of the convolutional layers and the associated computational cost quickly become limiting factors when increasing the voxelization resolution. In this work, the use of a dual-resolution network alleviates the problem, and the predictive power appears more

limited by the unbalanced dataset than by the voxelization resolution employed. Nevertheless, directly encoding and learning interfaces as cloud points[72,73] or meshes[74,75] appears more natural. However, such nonuniform representations cannot directly leverage the convolutional paradigm, unless its basic operations (convolution, pooling...) are completely redefined. For instance, Hanocka et al use a transformation-invariant encoding of adjacent edges, which can be collapsed to emulate pooling.[74] These techniques are still experimental and their pros and cons not as well mastered as traditional convolutional networks. Another possibility for improvement would be to encode coevolution data[76] inside interface voxels. Enhancing structural data with coevolution information has already proven successful,[19] and could yield even better results with this work's simple yet powerful representation of PP interfaces for the prediction of molecular function.

Conclusion

By using convolutional deep learning, this study demonstrates how a simple discretized representation can preserve a meaningful proportion of the wealth of information contained within the global shape of PP interfaces. The use of convolutional techniques also naturally solves the problem of mixing local and global structural descriptors within the dataset. It is my hope that this study provides additional incentive to research novel interface topology encodings amenable to deep learning and to implement them within de novo design and/or function prediction pipelines.

Acknowledgements

The calculations presented herein were performed using HPC resources from the MatriCS computing platform of Université de Picardie - Jules Verne, Amiens, France.

Data and Software Availability

The complete dataset, consisting for each PP interface of (i) the 3D mesh of the Voronoi interface and its fine and coarse voxelizations and (ii) the associated labels (number of α and β motifs and gene ontology terms), as well as the source code implementing the neural networks, are available for download at https://extra.u-picardie.fr/nextcloud/index.php/s/sn7NttiFrp9EcmY (warning: the complete uncompressed dataset files are close to 300 gigabytes in size).

Supporting Information Available

Dataset generation flowchart; statistics on selected molecular function GOT sets and justification for the retained set; demonstration of the inadequacy of accuracy for unbalanced multiclass problems; evolution of network performance along the training process; comparison of Pfam and GOT labeling within ProtCID clusters relevant to this study; structural similarity scores within the clusters of convolutional filter activators.

References

(1) Bruzzoni-Giovanelli, H.; Alezra, V.; Wolff, N.; Dong, C. Z.; Tuffery, P.; Rebollo, A. Interfering Peptides Targeting Protein-Protein Interactions: The Next Generation of Drugs? *Drug Discov. Today* **2018**, *23*, 272–285.

(2) Kerkhofs, M.; Bultynck, G.; Vervliet, T.; Monaco, G. Therapeutic Implications of Novel Peptides Targeting ER-Mitochondria Ca^{2+}-Flux Systems. *Drug Discov. Today* **2019**, *24*, 1092–1103.

(3) Lu, S.; Jang, H.; Gu, S.; Zhang, J.; Nussinov, R. Drugging Ras GTPase: A Comprehensive Mechanistic and Signaling Structural View. *Chem. Soc. Rev.* **2016**, *45*, 4929–4952.

(4) Milroy, L.-G.; Grossmann, T. N.; Hennig, S.; Brunsveld, L.; Ottmann, C. Allosteric Modulators of Protein-Protein Interactions. *Chem. Rev.* **2014**, *114*, 4695–4748.

(5) Cossar, P. J.; Lewis, P. J.; McCluskey, A. Protein-Protein Interactions as Antibiotic Targets: A Medicinal Chemistry Perspective. *Med. Res. Rev.* **2020**, *40*, 469–494.

(6) Ngo, T. D.; Plé, S.; Thomas, A.; Barette, C.; Fortuné, A.; Bouzidi, Y.; Fauvarque, M. O.; Pereira De Freitas, R.; Francisco Hilário, F.; Attreé, I.; Wong, Y. S.; Faudry, E. Chimeric Protein-Protein Interface Inhibitors Allow Efficient Inhibition of Type III Secretion Machinery and Pseudomonas Aeruginosa Virulence. *ACS Infect. Dis.* **2019**, *5*, 1843–1854.

(7) Schoeters, F.; Van Dijck, P. Protein-Protein Interactions in Candida Albicans. *Front. Microbiol.* **2019**, *10*, 1–16.

(8) Carro, L. Protein-Protein Interactions in Bacteria: A Promising and Challenging Avenue Towards the Discovery of New Antibiotics. *Beilstein J. Org. Chem.* **2018**, *14*, 2881–2896.

(9) Moreira, I. S.; Fernandes, P. A.; Ramos, M. J. Hot Spots–A Review of the Protein-Protein Interface Determinant Amino-Acid Residues. *Proteins* **2007**, *68*, 803–812.

(10) Rickard, M. M.; Zhang, Y.; Gruebele, M.; Pogorelov, T. V. In-Cell Protein-Protein Contacts: Transient Interactions in the Crowd. *J. Phys. Chem. Lett.* **2019**, *10*, 5667–5673.

(11) Berg, A.; Peter, C. Simulating and Analysing Configurational Landscapes of Protein-Protein Contact Formation. *Interface Focus* **2019**, *9*, 20180062.

(12) Wang, X.; Yu, B.; Ma, A.; Chen, C.; Liu, B.; Ma, Q. Protein-Protein Interaction Sites Prediction by Ensemble Random Forests with Synthetic Minority Oversampling Technique. *Bioinformatics* **2019**, *35*, 2395–2402.

(13) Upadhyayula, R. S. Computational Investigation of Structural Interfaces of Protein Complexes with Short Linear Motifs. *J. Proteome Res.* **2020**, *19*, 3254–3263.

(14) Preto, A. J.; Matos-Filipe, P.; de Almeida, J. G.; Mourão, J.; Moreira, I. S. *Methods in Molecular Biology*; Springer, 2020; pp 267–288.

(15) Zeng, M.; Zhang, F.; Wu, F.-X.; Li, Y.; Wang, J.; Li, M. Protein–Protein Interaction Site Prediction through Combining Local and Global Features with Deep Neural Networks. *Bioinformatics* **2019**, *36*, 1114–1120.

(16) Wang, S.; Sun, S.; Li, Z.; Zhang, R.; Xu, J. Accurate De Novo Prediction of Protein Contact Map by Ultra-Deep Learning Model. *PLOS Comput. Biol.* **2017**, *13*, e1005324.

(17) Wang, X.; Terashi, G.; Christoffer, C. W.; Zhu, M.; Kihara, D. Protein Docking Model Evaluation by 3D Deep Convolutional Neural Networks. *Bioinformatics* **2020**, *36*, 2113–2118.

(18) Jones, D. T.; Kandathil, S. M. High Precision in Protein Contact Prediction Using Fully Convolutional Neural Networks and Minimal Sequence Features. *Bioinformatics* **2018**, *34*, 3308–3315.

(19) Liu, Y.; Palmedo, P.; Ye, Q.; Berger, B.; Peng, J. Enhancing Evolutionary Couplings with Deep Convolutional Neural Networks. *Cell Syst.* **2018**, *6*, 65–74.

(20) Qiao, Y.; Xiong, Y.; Gao, H.; Zhu, X.; Chen, P. Protein-Protein Interface Hot Spots Prediction Based on a Hybrid Feature Selection Strategy. *BMC Bioinformatics* **2018**, *19*, 1–16.

(21) Wang, W.; Yang, Y.; Wang, X.; Wang, W.; Li, J. Development of Convolutional Neural Network and Its Application in Image Classification: A Survey. *Opt. Eng.* **2019**, *58*, 040901.

(22) Gezawa, A. S.; Zhang, Y.; Wang, Q.; Yunqi, L. A Review on Deep Learning Approaches for 3D Data Representations in Retrieval and Classifications. *IEEE Access* **2020**, *8*, 57566–57593.

(23) Townshend, R. J. L.; Bedi, R.; Suriana, P. A.; Dror, R. O. End-to-End Learning on 3D Protein Structure for Interface Prediction. *arXiv* **2018**, arXiv:1807.01297.

(24) Amidi, A.; Amidi, S.; Vlachakis, D.; Megalooikonomou, V.; Paragios, N.; Zacharaki, E. I. EnzyNet: Enzyme Classification Using 3D Convolutional Neural Networks on Spatial Representation. *PeerJ* **2018**, *6*, e4750.

(25) Bouvier, B.; Grünberg, R.; Nilges, M.; Cazals, F. Shelling the Voronoi Interface of Protein-Protein Complexes Reveals Patterns of Residue Conservation, Dynamics, and Composition. *Proteins* **2009**, *76*, 677–692.

(26) Li, Z.; He, Y.; Wong, L.; Li, J. Progressive Dry-Core-Wet-Rim Hydration Trend in a Nested-Ring Topology of Protein Binding Interfaces. *BMC Bioinformatics* **2012**, *13*, 51.

(27) Xu, Q.; Dunbrack, R. L. ProtCID: A Data Resource for Structural Information on Protein Interactions. *Nat. Commun.* **2020**, *11*.

(28) Bullock, B. N.; Jochim, A. L.; Arora, P. S. Assessing Helical Protein Interfaces for Inhibitor Design. *J. Am. Chem. Soc.* **2011**, *133*, 14220–14223.

(29) Huang, P.-S.; Love, J. J.; Mayo, S. L. A De Novo Designed Protein-Protein Interface. *Protein Sci.* **2007**, *16*, 2770–2774.

(30) Nerattini, F.; Tubiana, L.; Cardelli, C.; Bianco, V.; Dellago, C.; Coluzza, I. Design of Protein-Protein Binding Sites Suggests a Rationale for Naturally Occurring Contact Areas. *J. Chem. Theory Comput.* **2018**, *15*, 1383–1392.

(31) Taechalertpaisarn, J.; Lyu, R. L.; Arancillo, M.; Lin, C. M.; Perez, L. M.; Ioerger, T. R.; Burgess, K. Correlations Between Secondary Structure- and Protein-Protein Interface-Mimicry: The Interface Mimicry Hypothesis. *Org. Biomol. Chem.* **2019**, *17*, 3267–3274.

(32) Petrey, D.; Chen, T. S.; Deng, L.; Garzon, J. I.; Hwang, H.; Lasso, G.; Lee, H.; Silkov, A.; Honig, B. Template-Based Prediction of Protein Function. *Curr. Opin. Struct. Biol.* **2015**, *32*, 33–38.

(33) Dey, F.; Zhang, Q. C.; Petrey, D.; Honig, B. Toward a "Structural BLAST": Using Structural Relationships to Infer Function. *Protein Sci.* **2013**, *22*, 359–366.

(34) Sawyer, N.; Watkins, A. M.; Arora, P. S. Protein Domain Mimics as Modulators of Protein-Protein Interactions. *Acc. Chem. Res.* **2017**, *50*, 1313–1322.

(35) Webb, B.; Sali, A. Comparative Protein Structure Modeling Using MODELLER. *Curr. Protoc. Bioinformatics* **2016**, *54*, 5.6.1–5.6.37.

(36) Grubmüller, H.; Groll, V.; Tavan, P. Solvate. 2010; `https://www.mpibpc.mpg.de/grubmueller/solvate`, accessed 06/04/2021.

(37) Cazals, F.; Dreyfus, T. The Structural Bioinformatics Library: Modeling in Biomolecular Science and Beyond. *Bioinformatics* **2017**, *33*, 997–1004.

(38) Ashburner, M.; Ball, C. A.; Blake, J. A.; Botstein, D.; Butler, H.; Cherry, J. M.; Davis, A. P.; Dolinski, K.; Dwight, S. S.; Eppig, J. T.; Harris, M. A.; Hill, D. P.; Issel-Tarver, L.; Kasarskis, A.; Lewis, S.;

Matese, J. C.; Richardson, J. E.; Ringwald, M.; Rubin, G. M.; Sherlock, G. Gene Ontology: Tool for the Unification of Biology. *Nat. Genet.* **2000**, *25*, 25–29.

(39) Carbon, S.; Douglass, E.; Dunn, N.; Good, B.; Harris, N. L.; Lewis, S. E.; Mungall, C. J.; Basu, S.; Chisholm, R. L.; Dodson, R. J.; Hartline, E.; Fey, P.; Thomas, P. D.; Albou, L. P.; Ebert, D.; Kesling, M. J.; Mi, H.; Muruganujan, A.; Huang, X.; Poudel, S.; Mushayahama, T.; Hu, J. C.; LaBonte, S. A.; Siegele, D. A.; Antonazzo, G.; Attrill, H.; Brown, N. H.; Fexova, S.; Garapati, P.; Jones, T. E.; Marygold, S. J.; Millburn, G. H.; Rey, A. J.; Trovisco, V.; Dos Santos, G.; Emmert, D. B.; Falls, K.; Zhou, P.; Goodman, J. L.; Strelets, V. B.; Thurmond, J.; Courtot, M.; Osumi, D. S.; Parkinson, H.; Roncaglia, P.; Acencio, M. L.; Kuiper, M.; Lreid, A.; Logie, C.; Lovering, R. C.; Huntley, R. P.; Denny, P.; Campbell, N. H.; Kramarz, B.; Acquaah, V.; Ahmad, S. H.; Chen, H.; Rawson, J. H.; Chibucos, M. C.; Giglio, M.; Nadendla, S.; Tauber, R.; Duesbury, M. J.; Del, N. T.; Meldal, B. H.; Perfetto, L.; Porras, P.; Orchard, S.; Shrivastava, A.; Xie, Z.; Chang, H. Y.; Finn, R. D.; Mitchell, A. L.; Rawlings, N. D.; Richardson, L.; Sangrador-Vegas, A.; Blake, J. A.; Christie, K. R.; Dolan, M. E.; Drabkin, H. J.; Hill, D. P.; Ni, L.; Sitnikov, D.; Harris, M. A.; Oliver, S. G.; Rutherford, K.; Wood, V.; Hayles, J.; Bahler, J.; Lock, A.; Bolton, E. R.; De Pons, J.; Dwinell, M.; Hayman, G. T.; Laulederkind, S. J.; Shimoyama, M.; Tutaj, M.; Wang, S. J.; D'Eustachio, P.; Matthews, L.; Balhoff, J. P.; Aleksander, S. A.; Binkley, G.; Dunn, B. L.; Cherry, J. M.; Engel, S. R.; Gondwe, F.; Karra, K.; MacPherson, K. A.; Miyasato, S. R.; Nash, R. S.; Ng, P. C.; Sheppard, T. K.; Shrivatsav Vp, A.; Simison, M.; Skrzypek, M. S.; Weng, S.; Wong, E. D.; Feuermann, M.; Gaudet, P.; Bakker, E.; Berardini, T. Z.; Reiser, L.; Subramaniam, S.; Huala, E.; Arighi, C.; Auchincloss, A.; Axelsen, K.; Argoud, G. P.; Bateman, A.; Bely, B.; Blatter, M. C.; Boutet, E.; Breuza, L.; Bridge, A.; Britto, R.; Bye-A-Jee, H.; Casals-Casas, C.; Coudert, E.; Estreicher, A.; Famiglietti, L.; Garmiri, P.; Georghiou, G.; Gos, A.; Gruaz-Gumowski, N.; Hatton-Ellis, E.; Hinz, U.; Hulo, C.; Ignatchenko, A.; Jungo, F.; Keller, G.; Laiho, K.; Lemercier, P.; Lieberherr, D.; Lussi, Y.; MacDougall, A.; Magrane, M.; Martin, M. J.; Masson, P.; Natale, D. A.; Hyka, N. N.; Pedruzzi, I.; Pichler, K.; Poux, S.; Rivoire, C.; Rodriguez-Lopez, M.; Sawford, T.; Speretta, E.; Shypitsyna, A.; Stutz, A.; Sundaram, S.; Tognolli, M.; Tyagi, N.; Warner, K.; Zaru, R.; Wu, C.; Chan, J.; Cho, J.; Gao, S.; Grove, C.; Harrison, M. C.; Howe, K.; Lee, R.; Mendel, J.; Muller, H. M.; Raciti, D.; Van Auken, K.; Berriman, M.; Stein, L.; Sternberg, P. W.; Howe, D.; Toro, S.; Westerfield, M. The Gene Ontology Resource: 20 Years and Still GOing Strong. *Nucleic Acids Res.* **2019**, *47*, D330–D338.

(40) Gilpin, W. PyPDB: A Python API for the Protein Data Bank. *Bioinformatics* **2016**, *32*, 159–160.

(41) Rada, R.; Mili, H.; Bicknell, E.; Blettner, M. Development and Application of a Metric on Semantic Nets. *IEEE Trans. Syst. Man Cyb.* **1989**, *19*, 17–30.

(42) Gong, Y.; Jia, Y.; Leung, T. K.; Toshev, A.; Ioffe, S. Deep Convolutional Ranking for Multilabel Image Annotation. *arXiv* **2013**, arXiv:1312.4894.

(43) Wang, S.; Ma, J.; Peng, J.; Xu, J. Protein Structure Alignment Beyond Spatial Proximity. *Sci. Rep.* **2013**, *3*.

(44) Hinton, G. E. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507.

(45) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., D'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc., 2019; pp 8024–8035.

(46) Van Der Walt, S.; Schönberger, J. L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J. D.; Yager, N.; Gouillart, E.; Yu, T. Scikit-Image: Image Processing in Python. *PeerJ* **2014**, *2014*, 1–18.

(47) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

(48) Gao, X.; Xiao, B.; Tao, D.; Li, X. A Survey of Graph Edit Distance. *Pattern Anal. Appl.* **2010**, *13*, 113–129.

(49) Dobson, P. D.; Doig, A. J. Predicting Enzyme Class from Protein Structure Without Alignments. *J. Mol. Biol.* **2005**, *345*, 187–199.

(50) Kumar, C.; Choudhary, A. A Top-Down Approach to Classify Enzyme Functional Classes and Sub-Classes Using Random Forest. *EURASIP J. Bioinf. Syst. Biol.* **2012**, *2012*, 1–14.

(51) Erhan, D.; Bengio, Y.; Courville, A.; Vincent, P. *Technical Report 1341: Visualizing Higher-Layer Features of a Deep Network*; University of Montréal, 2009; pp 1–13.

(52) Gatys, L. A.; Ecker, A. S.; Bethge, M. Image Style Transfer Using Convolutional Neural Networks. 2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016; pp 2414–2423.

(53) Parui, S.; Jana, B. Relative Solvent Exposure of the Alpha-Helix and Beta-Sheet in Water Determines the Initial Stages of Urea and Guanidinium Chloride-Induced Denaturation of Alpha/Beta Proteins. *J. Phys. Chem. B* **2019**, *123*, 8889–8900.

(54) Holm, L. DALI and the Persistence of Protein Shape. *Protein Sci.* **2019**, *29*, 128–140.

(55) Zhang, Y. TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score. *Nucleic Acids Res.* **2005**, *33*, 2302–2309.

(56) Baspinar, A.; Cukuroglu, E.; Nussinov, R.; Keskin, O.; Gursoy, A. PRISM: A Web Server and Repository for Prediction of Protein–Protein Interactions and Modeling their 3D Complexes. *Nucleic Acids Res.* **2014**, *42*, W285–W289.

(57) Bateman, A.; Martin, M.-J.; Orchard, S.; Magrane, M.; Agivetova, R.; Ahmad, S.; Alpi, E.; Bowler-Barnett, E. H.; Britto, R.; Bursteinas, B.; Bye-A-Jee, H.; Coetzee, R.; Cukura, A.; Silva, A. D.; Denny, P.; Dogan, T.; Ebenezer, T.; Fan, J.; Castro, L. G.; Garmiri, P.; Georghiou, G.; Gonzales, L.; Hatton-Ellis, E.; Hussein, A.; Ignatchenko, A.; Insana, G.; Ishtiaq, R.; Jokinen, P.; Joshi, V.; Jyothi, D.; Lock, A.; Lopez, R.; Luciani, A.; Luo, J.; Lussi, Y.; MacDougall, A.; Madeira, F.; Mahmoudy, M.; Menchi, M.; Mishra, A.; Moulang, K.; Nightingale, A.; Oliveira, C. S.; Pundir, S.; Qi, G.; Raj, S.; Rice, D.; Lopez, M. R.; Saidi, R.; Sampson, J.; Sawford, T.; Speretta, E.; Turner, E.; Tyagi, N.; Vasudev, P.; Volynkin, V.; Warner, K.; Watkins, X.; Zaru, R.; Zellner, H.; Bridge, A.; Poux, S.; Redaschi, N.; Aimo, L.; Argoud-Puy, G.; Auchincloss, A.; Axelsen, K.; Bansal, P.;

Baratin, D.; Blatter, M.-C.; Bolleman, J.; Boutet, E.; Breuza, L.; Casals-Casas, C.; de Castro, E.; Echioukh, K. C.; Coudert, E.; Cuche, B.; Doche, M.; Dornevil, D.; Estreicher, A.; Famiglietti, M. L.; Feuermann, M.; Gasteiger, E.; Gehant, S.; Gerritsen, V.; Gos, A.; Gruaz-Gumowski, N.; Hinz, U.; Hulo, C.; Hyka-Nouspikel, N.; Jungo, F.; Keller, G.; Kerhornou, A.; Lara, V.; Mercier, P. L.; Lieberherr, D.; Lombardot, T.; Martin, X.; Masson, P.; Morgat, A.; Neto, T. B.; Paesano, S.; Pedruzzi, I.; Pilbout, S.; Pourcel, L.; Pozzato, M.; Pruess, M.; Rivoire, C.; Sigrist, C.; Sonesson, K.; Stutz, A.; Sundaram, S.; Tognolli, M.; Verbregue, L.; Wu, C. H.; Arighi, C. N.; Arminski, L.; Chen, C.; Chen, Y.; Garavelli, J. S.; Huang, H.; Laiho, K.; McGarvey, P.; Natale, D. A.; Ross, K.; Vinayaka, C. R.; Wang, Q.; Wang, Y.; Yeh, L.-S.; Zhang, J.; Ruch, P.; Teodoro, D. UniProt: The Universal Protein Knowledgebase in 2021. *Nucleic Acids Res.* **2020**, *49*, D480–D489.

(58) Richter, F.; Baker, D. *Synthetic Biology*; Elsevier, 2013; pp 101–122.

(59) Guharoy, M.; Chakrabarti, P. Secondary Structure Based Analysis and Classification of Biological Interfaces: Identification of Binding Motifs in Protein-Protein Interactions. *Bioinformatics* **2007**, *23*, 1909–1918.

(60) Keskin, O.; Nussinov, R. Similar Binding Sites and Different Partners: Implications to Shared Proteins in Cellular Pathways. *Structure* **2007**, *15*, 341–354.

(61) Pasco, M.; Dolain, C.; Guichard, G. *Comprehensive Supramolecular Chemistry II*; Elsevier, 2017; pp 89–125.

(62) Mistry, J.; Chuguransky, S.; Williams, L.; Qureshi, M.; Salazar, G. A.; Sonnhammer, E. L. L.; Tosatto, S. C. E.; Paladin, L.; Raj, S.; Richardson, L. J.; Finn, R. D.; Bateman, A. Pfam: The Protein Families Database in 2021. *Nucleic Acids Res.* **2020**, *49*, D412–D419.

(63) Das, S.; Orengo, C. A. Protein Function Annotation Using Protein Domain Family Resources. *Methods* **2016**, *93*, 24–34.

(64) Saha, T. K.; Katebi, A.; Dhifli, W.; Hasan, M. A. Discovery of Functional Motifs from the Interface Region of Oligomeric Proteins Using Frequent Subgraph Mining. *IEEE/ACM Trans. Comp. Biol. Bioinf.* **2019**, *16*, 1537–1549.

(65) Mishra, N. K.; Chang, J.; Zhao, P. X. Prediction of Membrane Transport Proteins and their Substrate Specificities Using Primary Sequence Information. *PLoS ONE* **2014**, *9*, e100278.

(66) Naveed, H.; Jackups, R.; Liang, J. Predicting Weakly Stable Regions, Oligomerization State, and Protein-Protein Interfaces in Transmembrane Domains of Outer Membrane Proteins. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 12735–12740.

(67) Kumaraswamy, S. K.; Sastr, P.; Ramakrishnan, K. Multi-Source Subnetwork-Level Transfer in CNNs Using FilterTrees. 2018 International Joint Conference on Neural Networks (IJCNN). 2018.

(68) Bau, D.; Zhou, B.; Khosla, A.; Oliva, A.; Torralba, A. Network Dissection: Quantifying Interpretability of Deep Visual Representations. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.

(69) Bogan, A. A.; Thorn, K. S. Anatomy of Hot Spots in Protein Interfaces. *J. Mol. Biol.* **1998**, *280*, 1–9.

(70) Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing Properties of Neural Networks. International Conference on Learning Representations. 2014.

(71) Odena, A.; Dumoulin, V.; Olah, C. Deconvolution and Checkerboard Artifacts. *Distill* **2016**, 00003.

(72) Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; Solomon, J. M. Dynamic Graph CNN for Learning on Point Clouds. *ACM Trans. Graphics* **2019**, *38*, 1–12.

(73) Qi, C. R.; Yi, L.; Su, H.; Guibas, L. J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. *arXiv* **2017**, arXiv:1706.02413.

(74) Hanocka, R.; Hertz, A.; Fish, N.; Giryes, R.; Fleishman, S.; Cohen-Or, D. MeshCNN: A Network with an Edge. *ACM Trans. Graphics* **2019**, *38*, 1–12.

(75) Groueix, T.; Fisher, M.; Kim, V. G.; Russell, B. C.; Aubry, M. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. *arXiv* **2018**, arXiv:1802.05384.

(76) Morcos, F.; Pagnani, A.; Lunt, B.; Bertolino, A.; Marks, D. S.; Sander, C.; Zecchina, R.; Onuchic, J. N.; Hwa, T.; Weigt, M. Direct-Coupling Analysis of Residue Coevolution Captures Native Contacts Across Many Protein Families. *Proc. Natl. Acad. Sci. USA* **2011**, *108*, E1293–E1301.

Graphical TOC Entry

Voxelized representations of protein-protein interfaces can leverage powerful convolutional deep learning techniques to predict elements of protein structure and function.