Graphic Encoding of Macromolecules for Efficient High-Throughput Analysis

Trilce Estrada
University of New Mexico
estrada@cs.unm.edu

Jeremy Benson
University of New Mexico
jeremybenson@cs.unm.edu

Hector Carrillo-Cabada
University of New Mexico
hcarrillo@cs.unm.edu

Asghar M. Razavi
Weill Cornell Medical College,
Cornell University
asr2013@med.cornell.edu

Michel A. Cuendet
Weill Cornell Medical College,
Cornell University
mac2109@med.cornell.edu

Harel Weinstein
Weill Cornell Medical College,
Cornell University
haw2002@med.cornell.edu

Ewa Deelman
University of Southern California
deelman@isi.edu

Michela Taufer
University of Tennessee at Knoxville
taufer@utk.edu

ABSTRACT

The function of a protein depends on its three-dimensional structure. Current approaches based on homology for predicting a given protein's function do not work well at scale. In this work, we propose a scalable and generalizable representation of proteins that explicitly encodes secondary and tertiary structure into fix-sized images. In addition, we present a neural network architecture that exploits our data representation to perform protein function prediction. We validate the effectiveness of our encoding method and the strength of our neural network architecture through a 5-fold cross validation over roughly 63 thousand images, achieving an accuracy of 80% across 8 distinct functional classes. Our novel approach of encoding and classifying proteins enables real-time processing during folding or other trajectory experiments, leading to high-throughput analysis.

ACM Reference format:
Trilce Estrada, Jeremy Benson, Hector Carrillo-Cabada, Asghar M. Razavi, Michel A. Cuendet, Harel Weinstein, Ewa Deelman, and Michela Taufer. 2018. Graphic Encoding of Macromolecules for Efficient High-Throughput Analysis. In *Proceedings of ACM-BCB, Washington, D.C. USA, August 2018 (ACM-BCB'18),* 10 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Proteins are macromolecules in charge of a wide variety of biological functions. They are composed of sequences of amino acids (also called residues) that form a chain of chemical bonds. Proteins interact with their environment and fold into a three-dimensional structure depending on their amino acid sequence. It is in this folded shape that proteins are able to interact with other proteins and molecules to perform their functions. Identifying the function of a given protein is not a trivial operation. Homology methods are among the more common techniques to predict protein functions. These methods require measuring the similitude of a protein with respect to a large database of known amino acid sequences and structures. The key idea supporting the use of homology methods is that proteins with similar sequences have similar functions. On the other hand, the main weakness of homology methods is that they do not scale with the number of proteins to be compared. Multiple sequence alignment is NP-complete [10], and structural alignment is an instance of the traditional three-dimensional graph matching problem, which is known to be NP-hard [13] (i.e., there is no known algorithm that can solve these problems in polynomial time $O(n^c)$, where c is a constant and n is the size of the input, which in this case is the number of proteins and their size). As the number of proteins increases over time (e.g., with the advancing of crystallography and NMR techniques), more scalable analysis techniques are needed to fully take advantage of the information embedded in new and existing proteins.

In an effort to find scalable methods for the identification of protein functions, we shall look at machine learning (ML) methods such as convolutional neural networks (CNNs) and deep learning (DL) [38] that are revolutionizing the way in which data is analyzed and processed in real time. In particular, these methods are becoming the de-facto techniques in computer vision and image processing, and they are solving previously open problems such as object recognition [35] with very high accuracy. As their popularity increases, deep learning methods are starting to be used for scientific applications, and structural biology is not the exception. However, as the function of a protein directly depends on its three-dimensional structure [32], computational approaches for protein function prediction, and more generally protein analysis based on ML are limited by the way in which proteins are represented. Inherent differences between proteins, such as length, location of structural motifs, and different folding conformations, are some of the challenges for representing proteins in a way that can be adequately handled by machine learning techniques.

ACM-BCB'18, Washington, D.C. USA

DOI: 10.1145/nnnnnnn.nnnnnnn

Having the ability to represent heterogeneous macromolecules such as proteins in a way that (1) exposes their structure (i.e., secondary and tertiary characteristics) and (2) can be processed efficiently, has the potential to disrupt the scale at which molecular analysis is done today. In this paper we propose such an encoding. Our approach is to represent structural and conformational information of macromolecules into a codified image that is to say, an encoded protein. We propose to transform the problem of protein analysis (i.e., function prediction and partial structural matching) into a more computationally tractable image pattern recognition problem. Specifically, our proposed approach encodes structural information embedded into a protein without the demand for inter-protein alignment required in homology-based studies. As such, we are able to exploit the advantages of structural representations and perform operations totally in parallel. Figure 1 shows a visual comparison between three structural protein representations described in Section 2 (i.e., 3D Cartesian atoms, multi-fold, and surface) against our proposed graphic encoding. Our resulting encoding opens the door for structural biologists to use powerful image processing and ML techniques to analyze very large macromolecular databases in an efficient, high-throughput way. Large scale analyses of this magnitude can be used to identify inter-molecular patterns that may signal function, interaction, and homology in a broader sense.

For this body of research, we develop our study as a proof of concept: first we design an image-processing methodology to graphically encode proteins and then we test the method to classify a large dataset of proteins into 8 functional classes. One important advantage of our approach is that once our classifier is trained, it can be used for both function prediction and trajectory analysis in large-scale molecular dynamics simulation. Specifically, our contributions in this paper are the following:

(1) A generalizable representation of macromolecules that explicitly encodes secondary structural motifs and their spatial characteristics within the molecule. This representation exposes inter- and intra-molecular structural patterns without having to perform protein alignments.
(2) An image processing system based on convolutional neural networks that is able to use our graphic structural representation and predict protein function with an average of 80.6% accuracy.

The remainder of the paper is organized as follows: Section 2 discusses related work. Section 3 introduces our novel macromolecular representation. Section 4 discuses and evaluates a system for protein function prediction that serves as a proof of concept to highlight the power of our encoding. Section 5 concludes the paper and presents future research directions.

2 BACKGROUND AND STATE OF THE PRACTICE

Proteins can be represented by a variety of ways, each with their own advantages and disadvantages with respect to preserving or exposing information for specific purposes. In this section, we present a brief summary of some standard formats and focus on their applicability to protein function prediction.

2.1 Sequence Representation

DNA, RNA, or proteins can be represented by their amino acid residues sequences: a succession of letters using letters GACT for DNA, GACU for RNA, and the one-letter codes for the 20 natural amino acids for proteins. A common technique to identify functional or structural relationships among proteins depends on aligning their sequences to find global or local shared motifs. Aligned sequences are usually represented through matrices, where each sequence corresponds to a row. Alignments can include gaps between columns to allow for local dissimilarities. Pairwise sequence alignment is usually performed using dynamic programming (e.g., Smith–Waterman algorithm [37], Needleman–Wunsch algorithm [2, 28] both with a time complexity of $O(nm)$ [25], where n and m are sequence length for a pair sequence alignment).

Even with very large sequences, it is relatively cheap to align millions of proteins using modern parallel methods [19]. Using sequence alignment for protein function prediction is based on the idea that proteins with similar sequences (homologous) share similar functions. However, this is not always true, and it has been argued [41] that sequence alone is not enough for predicting protein functions and requires knowledge on the folding patterns of the protein's three-dimensional structure.

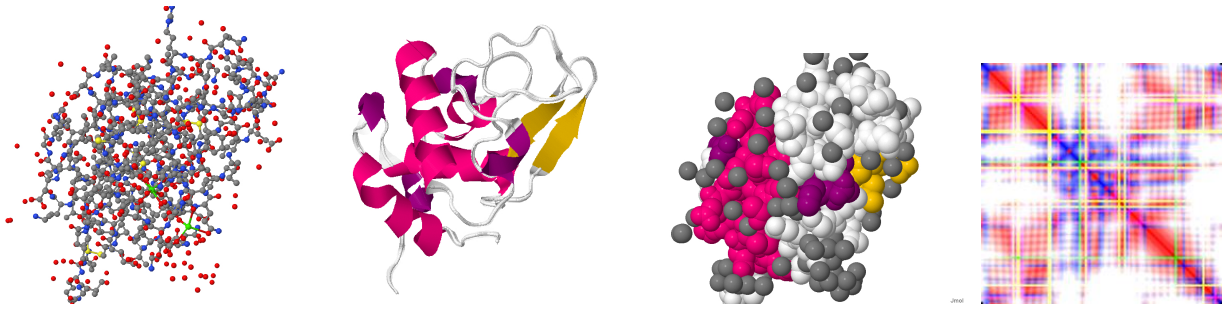
2.2 Structure Representations

Structural representations involve expressing, in a variety of ways, the three-dimensional arrangement of atoms in a protein. A 3D representation consists of the spatial coordinates of each of the (non-hydrogen) atoms in a Cartesian coordinate system. Alternatively, an angular representation expresses the protein's backbone conformation through its dihedral angles and their bond lengths (i.e., angles between planes of two sets of three atoms). With this representation, folds of proteins are expressed through the planes formed by four consecutive alpha Carbon atoms. Due to the degrees of freedom of both of these protein representations, their space complexity grows exponentially with the number of residues in the protein.

To deal with this complexity, the multi-fold representation is based on the observation that a protein's structure can be expressed through the combination of small structural units, called folding motifs, [5, 16, 17]. This representation takes advantage of collections of motifs that occur frequently and uses them as a meta-dictionary to express the entire protein's complexity in a condensed way. The most common representation of this kind uses folding motifs known as secondary structure motifs (e.g., helix, turn, and sheet). Methods for protein structure determination include X-ray crystallography, NMR spectroscopy, and electron microscopy [4].

Structural comparison and alignment of proteins is a critical aspect of multiple research problems, including protein annotation, and protein structure prediction. Structure-based function prediction often outperforms sequence-based methods because structural homologues contain similar folding patterns, even after evolution leads to their sequence similarity being completely undetectable [41].

Structural alignment combines sequence information with the secondary and tertiary structure of the protein or RNA molecule, and it is considered as the standard practice for homology-based



(a) 3D Cartesian representation (b) Multi-fold representation (c) Surface representation (d) Our graphic encoding

Figure 1: Visual depiction of multiple representations for the human alpha-lactalbumin protein (PDBid: 1A4V).

structure and function prediction [41]. But thoroughly comparing protein structures, whose size range from tens to several thousand amino acids, is computationally expensive, as three-dimensional matching is an NP-hard problem [13]. Moreover, for high-throughput analysis and identification of homologous structures, the alignment and comparison has to be done for multiple macromolecules at a time. As the alignment has to be done in a pair-wise manner and optimized globally, this process has limited opportunities for exploiting parallelism.

2.3 Other Protein Representations

There is a wide variety of other possible representations that are being used to describe proteins, their components, or binding pockets for example [27]. One such representation expresses only the molecular surface [26] as a set of functions (e.g., triangulations, polygons, distance distributions and landmark theory) on a unit sphere. This particular representation makes multiple protein comparison relatively easy [11, 18, 40], but it does not account for the internal structure of the protein, which is still crucial for determining the protein's functions.

Another representation treats the residues in a protein as if they were vectors in a 20-dimensional space [30]. In this case, a protein is represented as a random walk and proteins can be compared to each other through their vectorized profile. Ultimately, though, this representation loses its ability to express global folds or even protein domains in a way that can be used to characterize protein functions. The multipolar representation [14] offers a hierarchical parametric approach to characterizing the shape of a molecule. This representation uses multipoles (i.e., mathematical series that describe functions in terms of spherical harmonics) associated with coordinates of the Alpha Carbon of each residue as shape descriptors. The multipolar model reduces a protein to a vectorized format; calculating distances between proteins can be done through vector operations, rather than detailed alignment and spatial superimposition.

2.4 Deep Learning in Structural Biology

Machine learning and more recently deep learning have been used extensively in structural biology [7, 33, 42]. One of the main uses is in the prediction of secondary and tertiary structure of macromolecules. For example Li et al. [22] use a convolutional neural network with different kernel sizes to extract multi-scale features and predict secondary structure from protein sequences. Wang et al. [39] use two deep residual neural networks to perform contact prediction from protein sequences in order to improve folding accuracy.

Examples of more specific prediction problems include Hou et al. [15] using a deep convolution neural network (DeepSF) to classify a protein sequence into known folds, Nguyen et al.[29] proposing an ensemble of classifiers such as nearest neighbors, deep convolutional neural networks, and residual neural networks to predict a variety of angular and structural information with the final goal to predict loops, and Li et al. [21] compare the effectiveness of a deep neural network, a deep restricted Boltzmann machine, a deep recurrent neural network, and a deep recurrent restricted Boltzmann machine to predict phi and psi torsion angles of proteins' backbone.

More closely related to our work, deep learning has also been used for a variety of protein function prediction problems. Kulmanov et al. [20] propose DeepGO, a deep learning architecture used to learn features from protein sequences to predict function in the form of the Gene Ontology hierarchy. Similarly Liu et al. [24] use a recurrent neural network to predict four types of functions from protein sequences. In both cases, the neural network architecture is employed to form low-level feature representations from a simple input format as is the protein sequence. Cao et al. [6] propose ProLanGO, a deep recurrent neural network that deals with protein function prediction as if it was an analogous problem to language translation. This approach maps the protein sequence to a sequence of functions defined in the Gene Ontology.

Our work differs from all of the described related work in that we propose an encoding mechanism that captures secondary and tertiary information of proteins into an easy-to-analyze format. Our contribution is in the generalizable and homogeneous data representation, which can be used for multiple purposes, including for example in-situ analysis and indexing of folding trajectories, and protein function prediction is just one of such use cases. Note that this data representation is completely agnostic to the number of protein chains to be represented and the only change would be in the resolution of the image (i.e., the more residues are involved, the lower the resolution of the resulting image).

3 STRUCTURAL ENCODING OF MACROMOLECULES

In this section, we present our scalable graphic encoding of secondary and tertiary structure of proteins, while providing three key advantages over other structural representations:

- It is invariant to the protein size (i.e., number of amino acids). Proteins vary in size from tens to several thousand amino acids, still our graphic encoding can represent all of them using a standardized squared matrix.
- It exposes structural domains and folding motifs as patterns in an image.
- It can be built and queried in a fully parallel way.

Our encoding mechanism translates the complex structural and conformational information in three-dimensional proteins into a much simpler-to-analyze format: a three color channel image using a Red-Green-Blue (RGB) color model. The implicit advantage of this translation is in the ability of leveraging state-of-the-art image processing and ML mechanisms that currently cannot be used effectively with the canonical 3D or sequence representations of proteins. For our encoding technique, our final goal is to seamlessly expose local and global patterns from the protein's structural information without the need of pairwise structural alignments and homology calculations. The encoding process consists of four steps, also depicted in Figure 2 and explained in detail in the following subsections:

(1) Extracting secondary structural information using the Ramachandran plot.
(2) Expressing tertiary structural information via the distance matrix.
(3) Encoding secondary and tertiary information into multiple codified channels.
(4) Formatting the image (or tensor) into a fixed-size final encoding.

3.1 Extracting Secondary Structures with the Ramachandran Plot

The first step is to identify the basic molecular structures forming the protein. One way of doing this is through the analysis of backbone dihedral angles (angles between two intersecting planes that have two atoms in common) of the amino acid residues in the macromolecular structure. The Ramachandran [34] plot, originally developed by G. N. Ramachandran, C. Ramakrishnan, and V. Sasisekharan in 1963, determines the energetically allowable regions for the torsion angle phi, ϕ, (angle between the C-N-CA-C atoms) versus the torsion angle psi, ψ, (angle between the N-CA-C-N atoms), and omega ω (usually restricted to be 180 deg for the typical trans case or 0 deg for the rare cis case), for each residue of a protein sequence. Based on the constraints of the torsion angles (ϕ, ψ, and ω) as described by the Ramachandran, we can associate each amino acid residue in the protein with one of six types of secondary structures: α-helix, β-strand, Polyproline PII-helix, γ'-turn, γ-turn, and cis-peptide bonds.

3.2 Expressing Tertiary Structure through Distances

The second step seeks to establish a spatial correlation between the different secondary structures in the protein. In this step, we use the protein's distance matrix [31], which has been for example used as an aid to perform enzyme structural analysis and modeling [23]. For a protein with M Alpha Carbon atoms ($C\alpha$), its distance matrix is a squared matrix D of size $M \times M$, where the element in $D(i, j)$ corresponds to the distance between $C\alpha_i$ and $C\alpha_j$. Thus, making this a symmetric matrix. Note that the matrix is not restricted to a particular distance metric and we could use any metric or correlation coefficient for this purpose (e.g., Euclidean, squared Euclidean, Minkowsky, Chevychev, cosine, spearman, and hamming). For our experiments we choose to use the Euclidean distance between alpha carbon atoms in the backbone.

3.3 Encoding Secondary Structures to Multiple Color Channels

The third step combines the extracted secondary structures and distance matrix to represent the protein into a tensor. For practical purposes, and to take advantage of pre-built models for image processing, we decided to use a tensor of dimensions $M \times M \times 3$, where M is again the number of amino acid residues in the protein, and 3 indicates the Red-Green-Blue channels in an image. Thus, we use color to encode secondary structures, and intensity, or color saturation, to proportionally represent distances. Recall that in Step 1, amino acid residues were classified according to their dihedral angles into one of six secondary structures. Then, we can use the RGB model and six arbitrary colors to differentiate each structure as follows: α-helix, red; β-strand, blue; Polyproline PII-helix, magenta; γ'-turn, yellow; γ-turn, cyan; and cis-peptide bonds, green. If the structure could not be characterized by any of these six possibilities we use black. To encode a protein into its image representation, we define a function $sd(i, j)$, where $sd(i, j)$ is a normalized distance function that returns a value between 0 and 1 proportional to the distance between Alpha Carbon atoms i and j in the protein.

For a three channel image, and a particular residue identified as one of the seven possibilities (six secondary structures plus unidentified) we determine the color saturation of each channel according to the intended color assigned to that particular type of secondary structure. For example, for a red α-helix in position i, the saturation for channels red, green, and blue is $[1, sd(i, j), sd(i, j)]$ $\forall j \in D$. In the same way, the saturations for the other six structures in blue, magenta, yellow, cyan, green, and black are $[sd(i, j), sd(i, j), 1]$, $[1, sd(i, j), 1]$, $[1, 1, sd(i, j)]$, $[sd(i, j), 1, 1]$, $[sd(i, j), 1, sd(i, j)]$, $[0, 0, 0]$ respectively. This process is depicted in Figure 2. Note that even though we are building images, the number of channels that can be used are not restricted by three, besides of structure and distances, other information such as charge, and physical properties (e.g., hydrophobicity) could also be encoded into additional channels.

3.4 Formatting and Resizing

The final step consists of performing an image resizing (e.g., applying a bi-cubic interpolation) to produce an output of consistent dimensions across proteins regardless of their original length. Assuming a new size N the output is a $N \times N \times C$ tensor, where N is

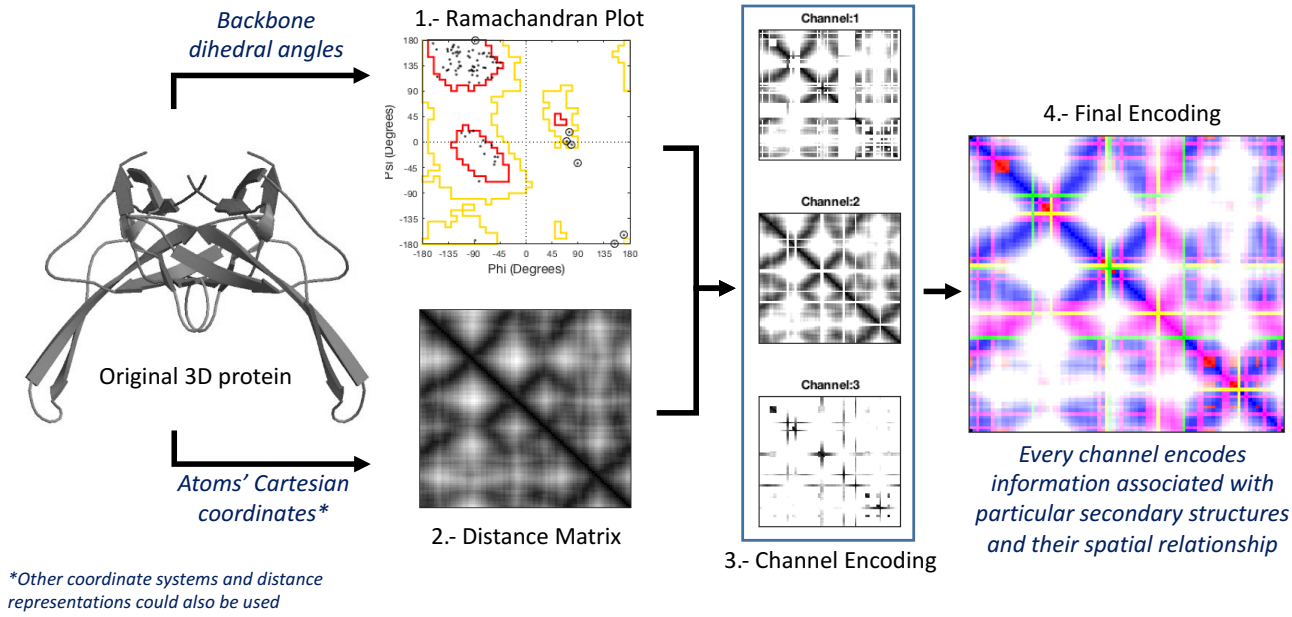


Figure 2: Example of encoding procedure for the gene V protein (PDBid: 1AE2).

the new size and could be smaller or larger than the original M, C is the number of channels used in the encoding (e.g., 3 channels). The output image, or tensor, either encodes more than one residue per pixel, or uses multiple pixels to encode one residue. The size of N can be chosen differently to optimize different performance metrics. For example, N can be equal to the number of amino acid residues in the longest protein in a dataset to optimize fidelity of the encoding; it can be the average number of residues in the dataset to keep a trade off between fidelity and efficiency; or it can be set to an arbitrary small size to maximize efficiency.

As our method provides a structural representation of proteins that is different from other formats, its analysis mechanisms are also different. Identifying structural motifs across a large database or performing protein modeling for function prediction does not require alignment and/or superimposition; thus, breaking a performance barrier for high throughput analysis. Figure 3 shows examples of some very different macromolecules in a three-dimensional representation and our graphic encoding. By looking at these images it is easy to distinguish how our encoding exposes patterns at different granularities in the image. Our representation transforms traditional structural biology problems into image pattern recognition, and it enables a straightforward use of sophisticated image processing and machine learning techniques for analysis and prediction.

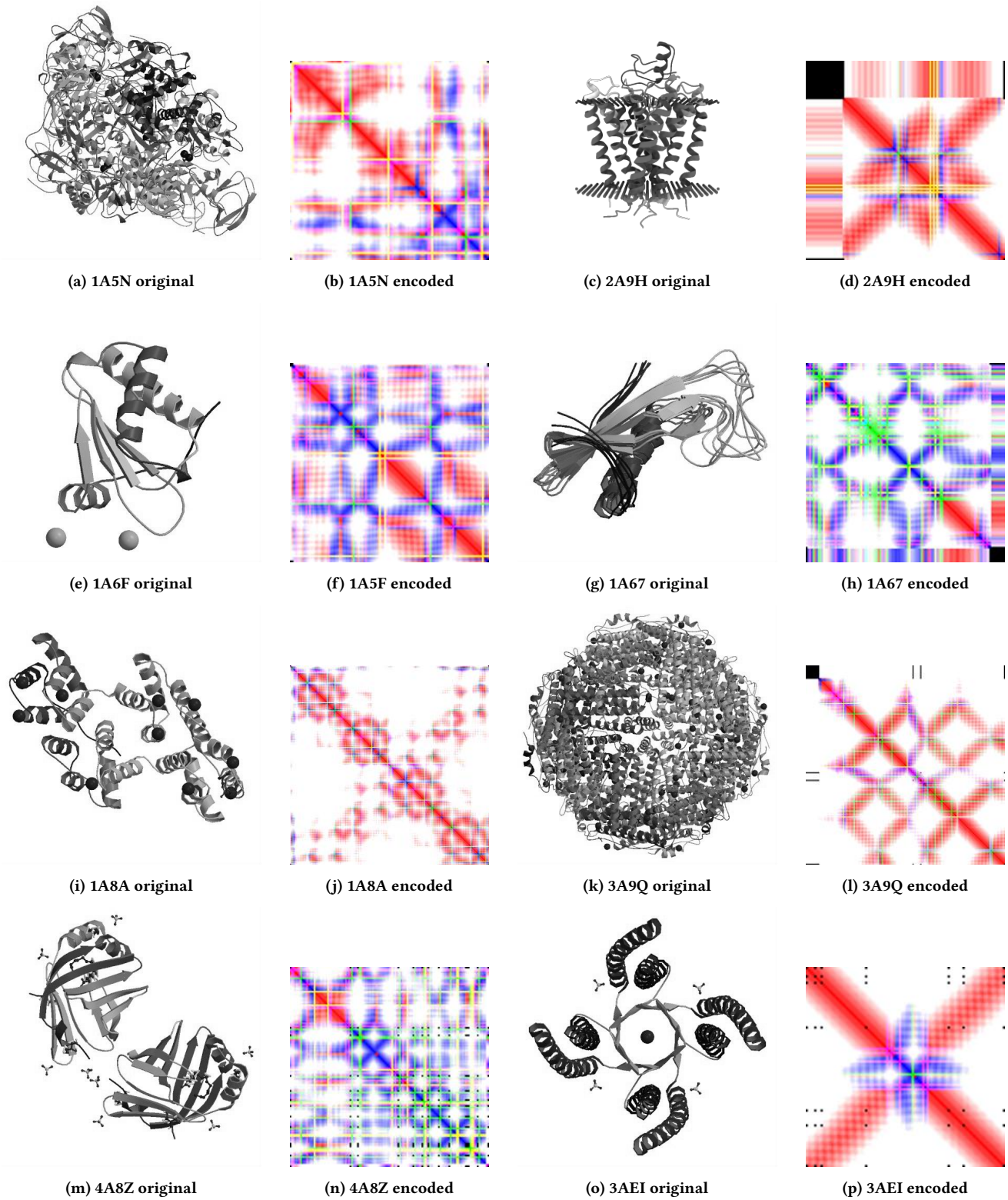
4 PROTEIN FUNCTION PREDICTION BASED ON IMAGES

Proteins contain a wide variety of structural motifs, which can also constitute functional microdomains that support the protein's functions. In this section we test the ability of our graphic encoding to expose local and global structural information necessary to perform basic protein function prediction.

4.1 Dataset Description

Our dataset consists of 62,991 proteins from the Protein Data Bank [3]. The protein data bank format (pdb) provides a standard representation for macromolecular structure data derived from X-ray diffraction and NMR studies. The file encodes a protein as a sequence of atoms, their type, and their three-dimensional coordinates. This representation can be easily converted to our encoding as explained in Section 3. Proteins in the dataset range in size from less than 100 non-hydrogen atoms to more than 50,000. The mean size is 6508 atoms with a standard deviation of 19495. Their mean resolution is 2.2 Angstroms, with a 1.7 standard deviation. The main source organism in this dataset is the *Homo Sapiens*, but the collection also includes *Escherichia coli*, *Mus musculus*, *Saccharomyces cerevisiae*, *Rattus norvegicus*, and *Mycobacterium tuberculosis* among others. Figure 3 depicts multiple examples of proteins in our dataset that were transformed from a three-dimensional structure to our graphic encoding.

To perform function prediction in this dataset, we obtain *GO terms* through the RCSB Protein Data Bank [4] and their biological details report. *GO terms* are established by the Gene Ontology Consortium [1, 8, 9] (GOC). GOC provides a standardized and consistent way of describing and documenting gene products across databases. The GO project comprises three structured ontologies with a well defined vocabulary to express gene product properties over three domains: cellular component, molecular function, and biological process in a species-independent manner. Terms in the cellular component describe the parts of a cell or its extracellular environment, for example a ribosome. Terms in the molecular function



(a) 1A5N original (b) 1A5N encoded (c) 2A9H original (d) 2A9H encoded

(e) 1A6F original (f) 1A5F encoded (g) 1A67 original (h) 1A67 encoded

(i) 1A8A original (j) 1A8A encoded (k) 3A9Q original (l) 3A9Q encoded

(m) 4A8Z original (n) 4A8Z encoded (o) 3AEI original (p) 3AEI encoded

Figure 3: Examples of macromolecular encodings for a diverse set of proteins.

describe activities that are performed by individual gene products or assembled complexes. Examples of such activities include binding or catalysis. Finally, terms identifying biological processes encompass series of events carried out by molecular function with a well defined beginning and end.

To label our dataset with specific functions, we use a biological process taxonomy provided by RCSB-PDB [4]. From this taxonomy we selected eight biological processes with the most number of proteins (i.e., more than 5,000) and use these groups as our classification targets. Table 1 describes this classification.

Table 1: Dataset breakdown in classes

Label	Function	GO-term	Number
0	Biological regulation	GO:0065007	5,241
Any process that modulates a measurable attribute of any biological process, quality or function.*			
1	Immune system process	GO:0002376	5,235
Any process involved in the development or functioning of the immune system, an organismal system for calibrated responses to potential internal or invasive threats.*			
2	Signaling	GO:0023052	7,242
The entirety of a process in which information is transmitted within a biological system. This process begins with an active signal and ends when a cellular response has been triggered.*			
3	Multi-organism process	GO:0051704	7,059
A biological process which involves another organism of the same or different species.*			
4	Catabolic process	GO:0009056	8,686
The chemical reactions and pathways resulting in the breakdown of substances, including the breakdown of carbon compounds with the liberation of energy for use by the cell or organism.*			
5	Localization	GO:0051179	5,727
Any process in which a cell, a substance, or a cellular entity, such as a protein complex or organelle, is transported, tethered to or otherwise maintained in a specific location. In the case of substances, localization may also be achieved via selective degradation.*			
6	Oxidation-reduction process	GO:0055114	11,026
A metabolic process that results in the removal or addition of one or more electrons to or from a substance, with or without the concomitant removal or addition of a proton or protons.*			
7	Biosynthetic process	GO:0009058	12,775
The chemical reactions and pathways resulting in the formation of substances; typically the energy-requiring part of metabolism in which simpler substances are transformed into more complex ones.*			

* *Description source http://amigo.geneontology.org*

The protein function classification is as follows: Label 0 contains proteins involved in biological regulation, this class is characterized by `GO:0065007` and contains 5,241 proteins. Label 1 is characterized by `GO:0002376`, indicative of immune system processes with 5,235 proteins. Label 2 is characterized by `GO:0023052` for signaling with 7,242 proteins. Label 3 is characterized by `GO:0051704` and represents multi-organism processes with 7,059 proteins. Label 4 contains 8,686 proteins involved in catabolic processes and is characterized by `GO:0009056`. Label 5 is characterized by `GO:0051179` for localization and contains 5,727 proteins. Label 6 is characterized by `GO:0055114` indicative of oxidation-reduction processes with 11,026 proteins. Finally, label 7 contains 12,775 proteins involved in biosynthetic processes characterized by `GO:0009058`.

4.2 Image classification

In recent years, increased computation power provided by general purpose graphic processing units (GPUs), the abundance of data, and better techniques to train and converge neural networks (e.g., activation and cost functions) have all given rise to Deep Learning solutions in every field, including structural biology [7, 33, 42]. Given enough computing power and data, deep neural architectures can build abstract representations capable of solving a wide variety of tasks in an end-to-end manner (i.e., without human intervention), replacing the traditional approach of carefully handcrafting features and algorithms. In particular, convolutional neural networks are becoming the state-of-the-art technique for classification, image processing, and computer vision.

4.2.1 Convolutional Neural Networks. A convolutional neural network, also known as a *CNN*, is a mathematical construction that trains complex non-linear functions out of linear compositions. CNNs handle matrix-oriented input, usually ingesting images, and produce a classification output. Convolutions are employed to preserve spatial relationships between pixels and learn important image features, such as edges, flattened areas, or other patterned shapes. A CNN is usually composed of a variety of convolutions (i.e., a filter kernel is convolved with an input), pooling (i.e., some input is down-sampled via some maximum or averaging over a neighborhood of pixels), and dense layers (i.e., a fully connected perception). Activation functions like sigmoid or rectified linear units help to remove noise, or smooth the data between layers. By representing secondary and tertiary structural information of proteins as 3D tensors, we seek to take advantage of CNN's superior capability in identifying spatial relationships, which in this context translates to finding structural patterns.

For one experiment in this work, we train a small version of VGG-net [36] from scratch. Shown to be successful for a variety of image classification tasks the VGG-net used here consists of 8 convolution and 3 fully connected layers, with very small receptive fields of 3×3 in layers that increase in width by a factor of two, starting from 64. The network also includes maxpooling layers right after each convolution layer.

4.2.2 Transfer Learning. Transfer Learning is the process of taking an existing neural network that has been trained on some dataset and re-purposing it for a new classification task. Specifically, the final layer of these networks are updated to handle new classes, but the convolutional filters learned from the initial training phase are kept, as they have learned to distinguish a vast feature space, finding notable differences like edges or unique patterns inherent to particular types of images.

Google's Inception-v3 [38] network is a general-purpose image recognition system trained for the ImageNet [35] large visual recognition challenge to discriminate entire images into 1,000 classes.

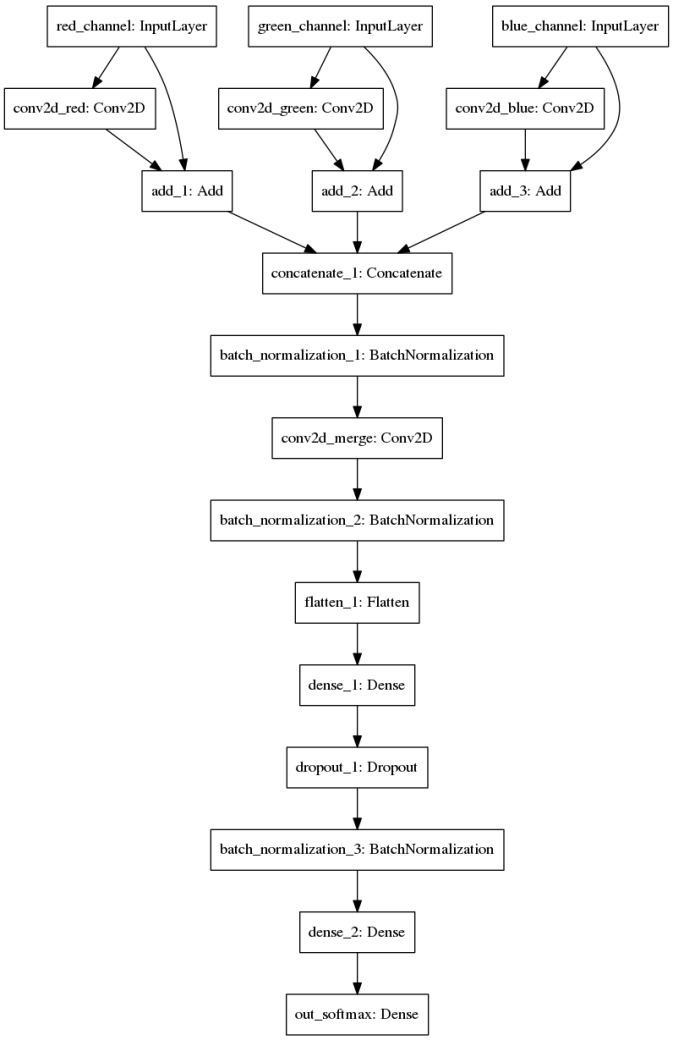


Figure 4: Our split-input resonant Graphic Encoding of Macromolecules Network used for classification, GEM-net

The architecture of the inception network is a series of inception modules, which are simply sets of convolutional filters that are concatenated together in order to capture information at varying kernel sizes (which is to say that, at each layer, the input is convolved with multiple kernels that vary in width and height; ultimately, the results of these convolutions are grouped together and sent to the next layer). To build a deep network of these characteristics with the hopes of it converging to a state that is practical for prediction typically requires a very large number of labeled images (i.e., the original Inception network for ImageNet was trained on 1.2 million images, with 50,000 images for validation and 100,000 images for testing [35]). The length of time the training phase takes is highly dependent on the compute capabilities of the machine. However, once these types of networks are trained, it is possible to take advantage of them in order to identify salient features from new classes; the network can be updated for a different classification task. This *transfer learning* harps on previous knowledge for a new task, without starting completely from scratch. Another pre-trained network available for retraining is MobileNet [12], a streamlined deep architecture designed for mobile and embedded systems.

In our preliminary experiments, we leveraged transfer learning by using both Inception-v3 and MobileNet. Although the images used to train these networks are significantly different from our protein dataset, the resulting classifiers are able to group images with reasonable accuracy (see Table 2: Results). These initial results indicate that our encoding method highlights the diverse features of the set, allowing these pre-trained networks to quickly and effectively triage the data. And while this first step showed encouraging class separation, it was clear that there was even room for improvement.

4.2.3 GEM-net: Graphic Encoding of Macromolecules Network. Noting that our encoding method proved useful for input into general purpose and pre-trained neural networks, we opted to develop an architecture that was specific for our tasks, in hopes that we would up performance. VGG-net and and the networks used in transfer learning all intake a 3-color channel image and apply convolutions and other operations directly. This immediate convolution means that the input channels are handled together, and in a sense, mashed together. However, in our encoding method, we particularly aim to maintain different pieces of information in the different color channels. It follows, then, that we should treat each channel independently, and also keep these inputs throughout the architecture (i.e., instead of perturbing the input and losing it, we could include it with subsequent layers).

Our Graphic Encoding of Macromolecules Network, or *GEM-net*, is a split-input resonant architecture designed to extract the most information from each channel, independently, and group the information thereafter. Figure **4** depicts the general architecture of GEM-net, in which we use a setup that first treats each color channel and then sends the combined tensor onward through a series of convolutional and fully connected layers. Batch normalization between layers serves to denoise the intermediate output tensors and help with both convergence time and final classification accuracy.

4.3 Evaluation

Using the neural network architectures described in the previous section, we evaluate over two encoding methods: the distance matrix of the protein (i.e, only information regarding tertiary structure encoded in one channel), and our proposed encoding mechanism consisting of three-channel images, where color represents secondary structures and saturation represents tertiary structure. Explicitly, to summarize, we train a VGG convolutional neural network from scratch, apply transfer learning to Inception and MobileNet, and work with our proposed architecture, GEM-net. For all of our tests we perform 5-fold cross validation, which splits the dataset into 5 disjoint partitions, each worth about 20% of the data. Then, training is done with 4 out of 5 partitions (i.e., 80%) and testing is done with the unseen partition. The process is repeated for a total of 5 times, using each time a different set of partitions for training and testing. Through this process, every protein in the dataset is used for training four times and for testing once. We use a learning rate of 0.005, which is standard for smaller size datasets. Batch

size of 100, and cross-entropy as our loss-function. The number of epochs we used varied per architecture.

The pretrained networks needed longer training periods because they only change weights in the last layer and use features they learned from general image classification in the other layers. The networks we trained from scratch converged quite quickly (within 10 epochs), further training steps only increased overfitting. Both data representations are square images of size 224. The hardware used for building our models is an Intel Xeon 8 core E5-1620 v4 at 3.50GHz and a GPU Tesla K80. A summary of our results is presented in Table 2, which presents performance metrics, including mean accuracy and training times in minutes.

We form the final class assignments based on their protein's *GO term* classification, as explained above. We note that none of this information is provided to our classifier. Like many convolutional architectures, the network relies solely on the images to learn distinguishing characteristics from the groups and perform a final classification.

Our results in Table 2 indicate that several of these image classifiers are able to discriminate among the eight classes of protein functions. The first thing to notice from the results is the added benefit of utilizing three channels of information (i.e., our proposed encoding) as opposed of just one (i.e., the distance matrix), as accuracy is consistently better.

Table 2: Results

Encoding: distance matrix			
Architecture	**Epochs**	**Accuracy**	**Training time**
MobileNet	100	21.01%	19 min.
Inception	100	33.45%	238 min.
MobileNet	500	36.82%	83 min.
Inception	500	39.00%	381 min.
VGG-net	10	18.26%	128 min.
GEM-net	10	69.73%	97min.
Encoding: proposed graphic representation			
Architecture	**Epochs**	**Accuracy**	**Training time**
MobileNet	100	24.11%	20 min.
Inception	100	36.28%	261 min.
MobileNet	500	44.15%	84 min.
Inception	500	47.54%	392 min.
VGG-net	10	21.02%	142 min.
GEM-net	10	**80.66%**	112 min.

Figure 5 shows a normalized confusion matrix for the prediction of eight classes using GEM-net and our graphic encoding. The matrix is a special instance of a contingency table that describes the performance of our classifier. Every row i represents instances whose correct classification is i. Every column j represents instances that were predicted as being of Class j. Cells in the diagonal indicate correct predictions. Every other cell indicates mistakes in the classification.

It is worth noting in our results that using our proposed encoding with GEM-net and only 10 epochs, prediction accuracy reaches above 85% for two of the classes (labels 6 and 7) and below 75% for only two of the classes (labels 0 and 5). One of the future directions

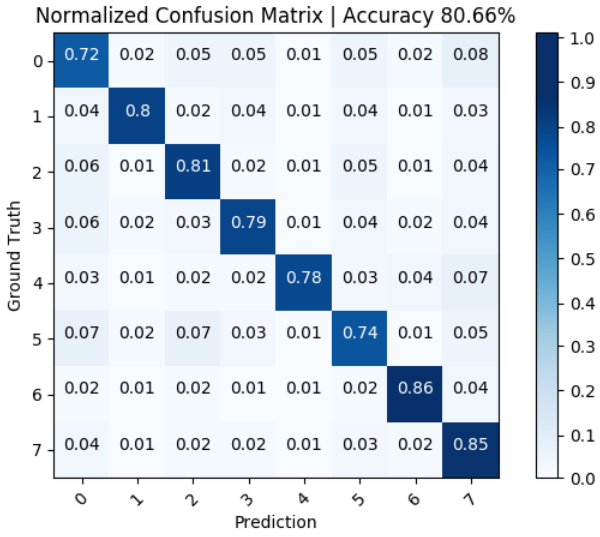


Figure 5: Confusion matrix for our protein classification.

for this work is to perform more focused function prediction (i.e., finer grained). A better-defined function is likely to be associated more tightly with one or several particular structural motifs. We expect that by looking at a narrower scope and a better defined biological function, the classifier will be able to achieve even better accuracy.

5 CONCLUSIONS AND FUTURE WORK

In this body of work, we present a generalizable and scalable approach to encode proteins that significantly boosts the capabilities of scientists seeking high-throughput techniques for the analytics of their ever-increasing molecular datasets. We also introduce a neural network architecture specifically geared towards analyzing proteins in this encoded format. The network relies on the idea of treating each color channel independently prior to grouping. This approach goes in line with our data representation, where each channel holds specific secondary structure information. Our approach does not rely on homology calculations and we can create these images in parallel, in addition to performing predictions concurrently. Our method can process a protein structure in few seconds, providing nearly instant feedback in, for example, in-situ analyses.

Ongoing work revolves around the idea of improving our classification by means of fine-tuning explicit operations (e.g., kernel strides and padding), as well as by taking an even closer look at the confidence of our prediction method (stored as probabilistic outcomes per class) through uncertainty quantification. For the future, we plan on integrating our preprocessing encoding method and classification models into protein folding simulations to analyze conformational changes at runtime on supercomputers.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation for the grants entitled *CAREER: Enabling Distributed and In-Situ Analysis for Multidimensional Structured Data* (NSF ACI-1453430) and *BIGDATA: IA: Collaborative Research: In Situ Data Analytics for Next Generation Molecular Dynamics Workflows* (NSF 1741057). We also thank the University of New Mexico Center for Advanced Research Computing for computational resources used in this work.

REFERENCES

[1] M. Ashburner, CA. Ball, JA. Blake, D. Botstein, H. Butler, JM. Cherry, AP. Davis, K. Dolinski, SS. Dwight, JT. Eppig, MA. Harris, DP. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, JC. Matese, JE. Richardson, M. Ringwald, GM. Rubin, and G. Sherlock. 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet.* 25, 1 (2000).

[2] David Barkan. 2002. A Parallel Implementation of the Needleman-Wunsch Algorithm for Global Gapped Pair-wise Alignment. *J. Comput. Sci. Coll.* 17, 6 (May 2002), 238–239. http://dl.acm.org/citation.cfm?id=775742.775778

[3] Helen Berman, Kim Henrick, and Haruki Nakamura. 2003. Announcing the worldwide Protein Data Bank. *Nature Structural Biology* 980, 10 (2003).

[4] Helen Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. 2000. Nucleic Acids Research. *Nature Structural Biology* 28, 1 (2000).

[5] Marenglen Biba, Floriana Esposito, Stefano Ferilli, Teresa M. A. Basile, and Nicola Di Mauro. 2007. *Multi-class Protein Fold Recognition Through a Symbolic-Statistical Framework.* Springer Berlin Heidelberg, Berlin, Heidelberg, 666–673.

[6] Renzhi Cao, Colton Freitas, Leong Chan, Miao Sun, Haiqing Jiang, and Zhangxin Chen. 2017. ProLanGO: Protein Function Prediction Using Neural Machine Translation Based on a Recurrent Neural Network. *arXiv:1710.07016 [cs, q-bio]* (Oct. 2017). http://arxiv.org/abs/1710.07016 arXiv: 1710.07016.

[7] Hongming Chen, Ola Engkvist, Yinhai Wang, Marcus Olivecrona, and Thomas Blaschke. 2018. The rise of deep learning in drug discovery. *Drug Discovery Today* (Jan. 2018). DOI: http://dx.doi.org/10.1016/j.drudis.2018.01.039

[8] The Gene Ontology Consortium. Gene Ontology Consortium. http://www.geneontology.org/. (????).

[9] The Gene Ontology Consortium. 2017. Expansion of the Gene Ontology knowledgebase and resources. *Nucleic Acids Res.* 4, 45 (2017).

[10] Isaac Elias. 2006. Settling the intractability of multiple alignment. *J Comput Biol* 13, 7 (2006), 1323–1339.

[11] Leif Ellingson and Jinfeng Zhang. 2011. An Efficient Algorithm for Matching Protein Binding Sites for Protein Function Prediction. In *Proceedings of the 2Nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine (BCB '11).* ACM, New York, NY, USA, 289–293. DOI: http://dx.doi.org/10.1145/2147805.2147837

[12] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. (04 2017).

[13] Michael R. Garey and David S. Johnson. 1990. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA.

[14] Apostol Gramada and Philip E. Bourne. 2006. Multipolar representation of protein structure. *BMC Bioinformatics* 67, 242 (2006).

[15] Jie Hou, Badri Adhikari, and Jianlin Cheng. 2018. DeepSF: deep convolutional neural network for mapping protein sequences to folds. *Bioinformatics* 34, 8 (April 2018), 1295–1303. DOI: http://dx.doi.org/10.1093/bioinformatics/btx780

[16] Jingtong Hou, Gregory E. Sims, Chao Zhang, and Sung-Hou Kim. 2002. A global representation of the protein fold space. *PNAS* 100, 5 (2002).

[17] Eugene Ie, Jason Weston, William Stafford Noble, and Christina Leslie. 2005. Multi-class Protein Fold Recognition Using Adaptive Codes. In *Proceedings of the 22Nd International Conference on Machine Learning (ICML '05).* ACM, New York, NY, USA, 329–336. DOI: http://dx.doi.org/10.1145/1102351.1102393

[18] Sungchul Kim, Sael Lee, and Hwanjo Yu. 2012. Indexing Methods for Efficient Protein 3D Surface Search. In *Proceedings of the ACM Sixth International Workshop on Data and Text Mining in Biomedical Informatics (DTMBIO '12).* ACM, New York, NY, USA, 41–48. DOI: http://dx.doi.org/10.1145/2390068.2390078

[19] N. Kolker, R. Higdon, W. Broomall, L. Stanberry, D. Welch, W. Lu, W. Haynes, R. Barga, and E. Kolker. 2011. Classifying proteins into functional groups based on all-versus-all BLAST of 10 million proteins. *OMICS* 15, 513 (2011).

[20] Maxat Kulmanov, Mohammed Asif Khan, Robert Hoehndorf, and Jonathan Wren. 2018. DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics* 34, 4 (Feb. 2018), 660–668. DOI: http://dx.doi.org/10.1093/bioinformatics/btx624

[21] Haiou Li, Jie Hou, Badri Adhikari, Qiang Lyu, and Jianlin Cheng. 2017. Deep learning methods for protein torsion angle prediction. *BMC Bioinformatics* 18 (Sept. 2017), 417. DOI: http://dx.doi.org/10.1186/s12859-017-1834-2

[22] Zhen Li and Yizhou Yu. 2016. Protein Secondary Structure Prediction Using Cascaded Convolutional and Recurrent Neural Networks *(IJCAI'16).* AAAI Press, New York, New York, USA, 2560–2567. http://dl.acm.org/citation.cfm?id=3060832.3060979

[23] Michael N. Liebman, Carol A. Venanzi, and Harel Weinstein. 1985. Structural analysis of carboxypeptidase A and its complexes with inhibitors as a basis for modeling enzyme recognition and specificity. *Biopolymers* 24, 9 (1985), 1721–1758.

[24] Xueliang Liu. 2017. Deep Recurrent Neural Network for Protein Function Prediction from Sequence. *arXiv:1701.08318 [cs, q-bio, stat]* (Jan. 2017). http://arxiv.org/abs/1701.08318 arXiv: 1701.08318.

[25] Saeed Maleki, Madanlal Musuvathi, and Todd Mytkowicz. 2016. Low-Rank Methods for Parallelizing Dynamic Programming Algorithms. *ACM Trans. Parallel Comput.* 2, 4, Article 26 (Feb. 2016), 32 pages. DOI: http://dx.doi.org/10.1145/2884065

[26] Richard J. Morris, Rafael J. Najmanovich, Abdullah Kahraman, and Janet M. Thornton. 2005. Real spherical harmonic expansion coefficients as 3D shape descriptors for protein binding pocket and ligand comparisons. *Bioinformatics* 21, 10 (2005).

[27] Yukari Nakamura, Ayaka Kaneko, and Takayuki Itoh. 2011. An Accelerated Pocket Extraction and Evaluation Technique for Druggability Analysis with Protein Surfaces. In *SIGGRAPH Asia 2011 Posters (SA '11).* ACM, New York, NY, USA, Article 31, 1 pages. DOI: http://dx.doi.org/10.1145/2073304.2073338

[28] Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology* 48, 3 (1970), 443 – 453. DOI: http://dx.doi.org/https://doi.org/10.1016/0022-2836(70)90057-4

[29] S. P. Nguyen, Z. Li, D. Xu, and Y. Shang. 2017. New Deep Learning Methods for Protein Loop Modeling. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2017), 1–1. DOI: http://dx.doi.org/10.1109/TCBB.2017.2784434

[30] M Novic and M Randic. 2008. Representation of proteins as walks in 20-D space. *SAR QSAR Environ Res* 19, 3 (2008).

[31] T. Ooi and K. Nishikawa. 1973. Conformation of Biological Molecules and Polymers. *E. D. and Pullman, B., Eds.* (1973), 173–187.

[32] Margarita Osadchy and Rachel Kolodny. 2011. Maps of protein structure space reveal a fundamental relationship between protein structure and function. *Biophysics and Computational Biology* 108, 30 (2011).

[33] Kuldip Paliwal, James Lyons, and Rhys Heffernan. 2015. A Short Review of Deep Learning Neural Networks in Protein Structure Prediction Problems. *Advanced Techniques in Biology & Medicine* 3, 3 (Sept. 2015), 1–2.

[34] G.N. Ramachandran, C. Ramakrishnan, and V. Sasisekharan. 1963. Multipolar representation of protein structure. *Journal of Molecular Biology* 7, 95 (1963).

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115, 3 (2015), 211–252. DOI: http://dx.doi.org/10.1007/s11263-015-0816-y

[36] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. (2014).

[37] T.F. Smith and M.S. Waterman. 1981. Identification of common molecular subsequences. *Journal of Molecular Biology* 147, 1 (1981), 195 – 197. DOI: http://dx.doi.org/https://doi.org/10.1016/0022-2836(81)90087-5

[38] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2015. Rethinking the Inception Architecture for Computer Vision. *CoRR* abs/1512.00567 (2015).

[39] Sheng Wang and Jinbo Xu. 2017. De Novo Protein Structure Prediction by Big Data and Deep Learning. *Biophysical Journal* 112, 3 (Feb. 2017), 55a. DOI: http://dx.doi.org/10.1016/j.bpj.2016.11.334

[40] Yong Wang, Wu Ling-Yun, Ji-Hong Zhang, Zhong-Wei Zhan, Zhang Xiang-Sun, and Chen Luonan. 2009. Evaluating Protein Similarity from Coarse Structures. *IEEE/ACM Trans. Comput. Biol. Bioinformatics* 6, 4 (Oct. 2009), 583–593. DOI: http://dx.doi.org/10.1109/TCBB.2007.70250

[41] J.C. Whisstock and A.M. Lesk. 2003. Prediction of protein function from protein sequence and structure. *Q Rev Biophys* 36, 3 (2003).

[42] Mengying Zhang, Qiang Su, Yi Lu, Manman Zhao, and Bing Niu. 2017. Application of Machine Learning Approaches for Protein-protein Interactions Prediction. *Medicinal Chemistry (Shariqah (United Arab Emirates))* 13, 6 (2017), 506–514. DOI: http://dx.doi.org/10.2174/1573406413666170522150940