

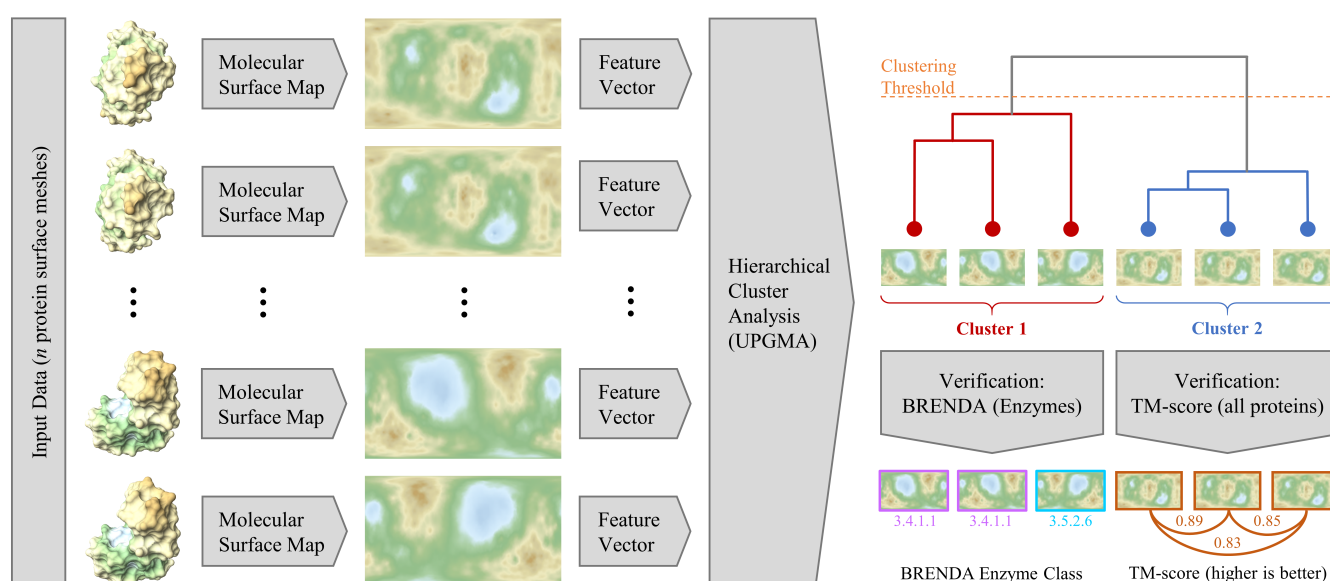
# Analyzing Protein Similarity by Clustering Molecular Surface Maps

Karsten Schatz<sup>1,\*</sup> , Florian Frieß<sup>1,\*</sup>, Marco Schäfer<sup>2</sup> , Thomas Ertl<sup>1</sup> , and Michael Krone<sup>2</sup> 

<sup>1</sup>Visualization Research Center, University of Stuttgart, Stuttgart, Germany

<sup>2</sup>Big Data Visual Analytics in Life Sciences, University of Tübingen, Tübingen, Germany

\*These authors contributed equally



**Figure 1:** Schematic overview of our hierarchical protein clustering approach (from left to right). The input is an ensemble of proteins. For each protein, a three-dimensional molecular surface representation is computed, which is subsequently transformed to a two-dimensional Molecular Surface Map [KSR\*17]. From each of these maps, a descriptive feature vector is extracted using either Image Moments, Color Moments, or a Convolutional Neural Network. Based on the distances between these feature vectors, a hierarchical clustering is computed using the UPGMA algorithm [SM58]. We verify the clustering results using either the BRENDA data base that provides classes of functionally similar enzymes, and the TM-score [ZS04], which provides a similarity measure for two proteins.

## Abstract

Many biochemical and biomedical applications like protein engineering or drug design are concerned with finding functionally similar proteins, however, this remains to be a challenging task. We present a new imaged-based approach for identifying and visually comparing proteins with similar function that builds on the hierarchical clustering of Molecular Surface Maps. Such maps are two-dimensional representations of complex molecular surfaces and can be used to visualize the topology and different physico-chemical properties of proteins. Our method is based on the idea that visually similar maps also imply a similarity in the function of the mapped proteins. To determine map similarity we compute descriptive feature vectors using image moments, color moments, or a Convolutional Neural Network and use them for a hierarchical clustering of the maps. We show that image similarity as found by our clustering corresponds to functional similarity of mapped proteins by comparing our results to the BRENDA database, which provides a hierarchical function-based annotation of enzymes. We also compare our results to the TM-score, which is a similarity value for pairs of arbitrary proteins. Our visualization prototype supports the entire workflow from map generation, similarity computing to clustering and can be used to interactively explore and analyze the results.

## CCS Concepts

• **Human-centered computing** → Dendrograms; Scientific visualization; • **Applied computing** → Bioinformatics;

## 1. Introduction

Understanding the relationship between protein structures remains a challenging yet important task for many application areas, like drug design or biomedical research. Finding similar proteins manually is a time-consuming task and it is not always clear how to compare proteins. Therefore, a large number of automatic clustering or similarity comparison methods have been developed. They either directly compare the sequence of amino acids forming the protein to find molecules with similar behavior, or use the spatial arrangement of the atoms to find proteins with similar shape. For sequence data, popular tools like BLAST [AGM\*90], or Clustal [LBB\*07] are available. Spatial comparisons use the atomic coordinates to find proteins with similar layout and size. An example working on unaligned proteins is 3D-Surfer 2.0 [XESK14], which computes a feature vector containing 3D Zernike Descriptors for finding similarities, which are invariant under rotation.

Molecular surfaces are generally considered to be influential for the function of a protein [KKF\*17], however, they are visually complex and suffer from occlusion, as all three-dimensional depictions. Krone et al. [KSR\*17] presented Molecular Surface Maps to solve the 3D occlusion problem and to reduce visual complexity. Additionally, they showed in their work that such two-dimensional representations of protein surfaces can be used for overview and comparison. The maps can show physico-chemical properties as well as the topography of the molecular surface in a two-dimensional representation.

We present a method that is based on the idea that visually similar maps also imply a similarity in the function of the underlying protein [BJ09; TL12]. Therefore, we propose a hierarchical clustering of Molecular Surface Maps using an image-based similarity score. We evaluate different structural and physico-chemical properties and analyze the resulting clustering by comparing it against established methods and databases, namely the TM-score [ZS04] and the BRENDA enzyme data base (see Figure 1).

We implemented our proposed method in a prototypical visualization application, which allows to compute an image-based clustering of the aforementioned Molecular Surface Maps and to verify, interactively explore, and analyze the results. The hierarchical clustering is visualized as a dendrogram, as shown in Figure 1. In order to create a unique descriptor used for the similarity clustering of the Molecular Surface Maps, we either compute Image Moments [Hu62; Flu00], Color Moments [MSM09], or use a Convolutional Neural Network [SHZ\*18] to extract features for each map. This descriptor allows us to identify similar maps even if the content is translated, rotated, or scaled. The distance between a pair of descriptors is used as a measure of similarity. Our prototype application provides different views that show the hierarchical clustering tree and can be used to show the three-dimensional molecular surface in order to visually verify the computed results. It is also able to overlay the classification provided by the BRENDA database or the TM-score onto our results for verification.

Our contributions can be summarized as follows:

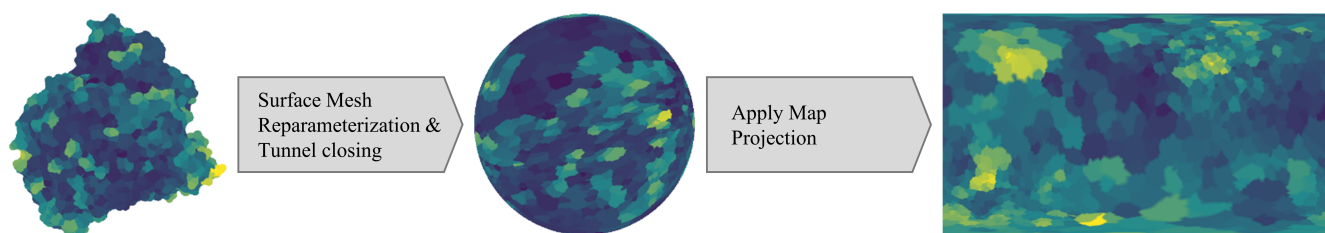
- We compare three different approaches for determining the similarity of Molecular Surface Maps (based on Image Moments, Color Moments, and the MobileNetV2 CNN).
- We tested different combinations of molecular surface properties for the protein similarity measure and evaluate the results using the enzyme classification provided by the BRENDA data base and the TM-score.
- We implemented a prototypical application that visualizes the hierarchical similarity-based clustering as a dendrogram and allows to visually explore the results using multiple linked views.


## 2. Biological Background

Proteins are macromolecules that consist of a single or multiple chains of amino acids. A typical chain has 150–500 amino acids, each one consisting of about 13–27 atoms. All amino acids have a basic structure in common, the *backbone* part, which consists of a central carbon atom called  $C\alpha$ , an amino group ( $NH_2$ ), a carboxyl group ( $COOH$ ), and a hydrogen (H). Besides the backbone, each amino acid has a so-called *side chain* that determines the individual chemical properties of this amino acid. In proteins, amino acids form chains via peptide bonds. These are characteristic covalent bonds that link the amino group of one amino acid to the carboxyl group the next one. That is, the chains of a protein have one end with an amino group (N-terminus) and one end with a carboxyl group (C-terminus). For more details, please refer to the book of Berg et al. [BTSC02]. When forming a protein, the chains fold into the so-called *tertiary structure*, the energetically most favorable three-dimensional conformation, which is held together mainly by hydrogen bonds. Proteins serve many different tasks in the bodies of all living creatures as well as in a wide variety of industrial and medical applications, thus their analysis is of great interest.

Starting from the amino acid sequence of a given protein, many predictions are possible. Using only this sequence, the tertiary structure and even the function of the complex can be inferred. At the widely-known CASP (Critical Assessment of protein Structure Prediction) experiment that takes place biennially a benchmark for all methods that try to predict the tertiary structure is often published [MPJF95]. Proteins with similar sequences often also share an evolutionary relationship and have similar function. To find proteins with similar sequences in a large data base, search methods such as BLAST [AGM\*90] (Basic Local Alignment Search Tool) have been developed. To express and to display the aforementioned evolutionary relationship and similarity, biologists typically use phylogenetic trees [FM67]. These trees are often rendered as cladograms (to display only relationships) or dendrograms (to additionally encode distance, see Figure 1). Dendrograms can be used to quickly find similar organisms in terms of structure, shape, or function, depending on the chosen comparison operator. However, proteins with vastly different sequences can have a similar three-dimensional structure due to the folding process. Therefore, it is possible that their function is also similar. This fact can be exploited for drug discovery [KW05], or to reveal distant evolutionary relationships between organisms [Koe01]. Conversely, some proteins can fold into different three-dimensional structures under certain circumstances. These misfolded proteins (called *prions*) can be dysfunctional, causing severe diseases like Alzheimer's or Creutzfeldt Jakob. That is, only the sequence of a protein is not always sufficient to faithfully analyze the function of a protein.

Chemically, the function of a protein is defined by its interface



**Figure 2:** Schematic overview of the steps involved in the generation of a Molecular Surface Map. From left to right: Solvent Excluded Surface, Molecular Surface Globe, and Molecular Surface Map. The protein (PDB ID: 1AJN) is colored by temperature factor (or b-factor) using the Viridis color map (  ) from matplotlib [Hun07]. The b-factor is an indicator of the flexibility of the protein.

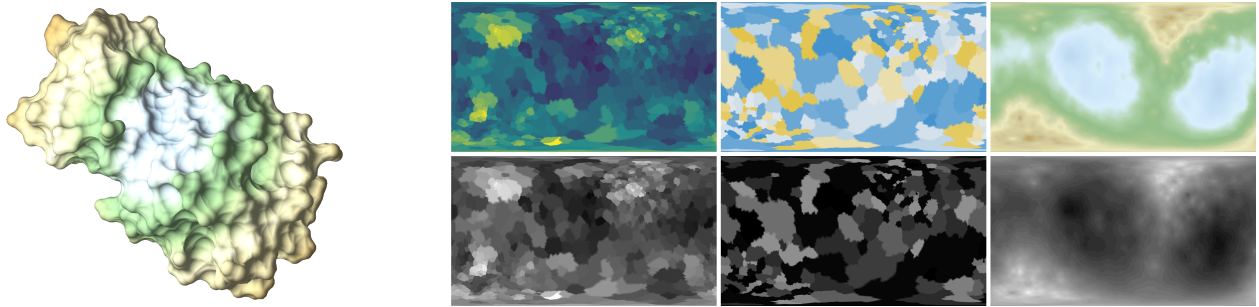
with the environment, that is, the shape and the physico-chemical properties of the parts of the protein that are accessible to potential reaction partners. To show this functional interface of a protein, several definitions for molecular surfaces have been developed. The most simple one is the van-der-Waals surface, where each atom is represented by a sphere with a radius equal to the van-der-Waals radius of the respective atom type. This surface mainly shows the shape and volume of a molecule, not its interface with respect to a reaction partner. Other surface definitions are the Solvent Accessible Surface (SAS) [LR71] and the Solvent Excluded Surface (SES) [Ric77; Con83]. These two methods are closely related, as they both generate the surface with respect to solvent molecules of a specific size. This solvent molecule is internally represented by a sphere of fixed radius (typically a value between 1.5 and 3.0 Å). The idea of both surface representations is to *roll* this solvent sphere over the van-der-Waals surface. The surface defined by the center of the sphere corresponds to the SAS, while the surfaces that the sphere is not able to reach because it is being blocked by the atoms corresponds to the SES. The SES is usually considered as more useful for a detailed analysis, since it intuitively shows the interface of a protein with respect to a certain smaller molecule like a solvent or ligand. That is, the SES is the surface that is reachable by this small molecule. In the past, numerous algorithms to compute the SES have been developed [KKF\*17]. Those algorithms can be split into ones that calculate an explicit mesh of the SES, such as MSMS by Sanner et al. [SOS96], or ones that render the SES directly without calculating an explicit representation beforehand (e.g., Krone et al. [KGE11] or Rau et al. [RZK\*19]). For visualization, all of the aforementioned surfaces can be enriched using color mapping to depict further functionally relevant properties of the protein. We use the SES as basis for our work, as it depicts the functionally relevant information, namely the shape and physico-chemical properties of the interface of a molecule and its environment.


### 3. Related Work

Map-like representations of proteins mostly base upon a spherical description of the geometry. Rahi and Sharp [RS07] re-parametrize the surface vertices of a molecule into spherical coordinates. This allows them to then map the surface onto a sphere in order to compare multiple proteins more easily. The actual comparison, however, is only a visual one and does not provide any countable distance measures. Building upon this approach, the Molecular Sur-

face Maps method of Krone et al. [KSR\*17] that is also used in this work further transforms those spherical descriptions by using conventional map projection methods generally used in cartography to generate two-dimensional maps of protein surfaces. Additionally, as opposed to other methods, occurring tunnels of the protein are closed before any projections to achieve an artifact-free final representation. Their results sometimes suffer from the drawbacks of the selected map projection method as there exists no mapping between a sphere and a map that is able to conserve area as well as distance perfectly. Hasegawa and Funatsu [HF12] use a spherical self-organizing map to achieve a projection from the protein surface onto a sphere. Due to their algorithm the results are not necessarily smooth and may contain holes that can get larger when the size of the projected protein decreases. Hass and Koehl [HK14] on the other hand use a conformal mapping to measure the roundness of a given protein. Opposed to our approach they do not incorporate further properties of the protein surface. In their Structuprint method, Kontopoulos et al. [KVTK16] project the protein onto a sphere by casting a straight line from the center point through the surface points. This approach therefore can lead to undesired overlaps of projected surface parts. Protein cavities or tunnels are also subject to mapping methods. Kolesár et al. [KBP\*16] unfold occurring tunnels into a map-like representation. They also use the Image Moments method to compare different tunnels, for which we will show later that it is not suitable for our needs. To further resemble the shape of the original cavity, Schatz et al. [SKB\*19] choose a hat-like shape as primitive, which then can be further simplified to a map in the form of a disk. While depicting the original shape better, this can lead to heavy distortions of the surface.

Apart from map projections, numerous methods have been developed to compare proteins among each other. La et al. and Xiong et al. [LEV\*09; XESK14] describe their 3D-Surfer software that uses 3D Zernike descriptors as described by Seal et al. [SLL\*08b] to extract feature vectors for each protein to achieve comparisons of larger protein numbers. In contrast to our work, they do not incorporate further biochemical properties of the surface that may be important to determine the function of a protein. Bock et al. [BGG07] describe an approach to compare areas on protein surfaces utilizing so-called spin images. Their computationally complex approach also utilizes only geometrical information but they suggest incorporating physico-chemical properties like in our approach. As proposed by Anzali et al. [ABK\*96], the shape of one molecule can



**Figure 3:** Overview of the generated scalar maps for the protein with the ID 6LU7 and their colorized versions. The first map shows the *b*-factor coloring, using the Viridis color map (  ), the second shows the hydrophobicity interpolated between yellow (hydrophobe), white and blue (hydrophil) and the last shows the cartography-inspired coloring by elevation, i. e. the distance in Å between the surface and the centroid of the protein. The gray value maps below show the scalar values used to generate the colors of the maps above. These values have been rescaled in order to make them visible.

be stored in a neural network to achieve direct comparison with others. Such direct comparisons are also possible through the use of gradient vector flow [SKR\*14], or, again by 3D Zernike descriptors [SLL\*08a]. All of the abovementioned methods have in common to be computationally relatively complex. Hofbauer et al. [HLA04] construct representatives in the form of graphs and then compare the resulting graphs among each other. They also compare physico-chemical properties, but their method, again, suffers from computational complexity, as the comparison between two proteins alone can take up to several minutes.

Besides purely biological application, the hierarchical clustering of images is also relevant for our work. A review of commonly used clustering algorithms has been given by Saxena et al. [SPG\*17]. The K-Means algorithm, probably first described by Steinhaus [Ste56], is one of the most well-known and widely used clustering algorithms. As it requires the knowledge of the number of clusters beforehand, it struggles to solve many real-world application cases like ours where this number is unknown. Nonetheless, it is used by the methods of Cai et al. [CHL\*04] and Pandey and Khanna [PK14]. The first method uses image features alongside with textual features to cluster images found in the world wide web. Such methods are not necessarily directly adaptable to biomedical images as the calculated features might differ heavily. The second method presents an approach similar to the one presented here. It agglomeratively clusters images by consecutively calculating representative images of the clusters and using them for further clustering. It is also intended for photographs and not biomedical datasets, and is thus not able to incorporate images with multiple physico-chemical properties.

One of the most well-known clustering applications in biology is the phylogenetic analysis. There, biologists try to understand the evolutionary relationships between different proteins or genes. The result of such a clustering are so-called phylogenetic trees [FM67], often visualized by a dendrogram. Several methods for their construction exist, the currently most widely used is the neighbor-joining method, first described by Saitou and Nei [SN87]. As opposed to the UPGMA method (Unweighted Pair Group with Arith-

metic Mean [SM58]) they do not necessarily produce rooted trees and do not assume a constant rate of evolution. Like all other methods constructing phylogenetic trees, the resulting visual layouts of these methods are often ambiguous, depending on the order of the input. We decided to use the UPGMA method in our approach, as it constructs rooted trees and the results can be interpreted more easily. To examine the differences in phylogenetic trees, Bremm et al [BvLH\*11] present a comparison approach. As the resulting tree structures become quite large, Huson et al. [HRR\*07] describe a visualization system for their efficient display.

#### 4. Algorithm

Our algorithm consists of four steps. First, the proteins are aligned according to their principle components. In the second step, three Molecular Surface Maps are created for all proteins in our input data set, each contains the scalar values of the associated surface property, see bottom row in Figure 3. The three properties are the topological structure of the surface encoded into a heightmap (as proposed by Krone et al. [KSR\*17]), the hydrophobicity, which describes how energetically (un-)favorable a direct interaction with water molecules would be, and the temperature factor (or *b*-factor), which is an indicator for the flexibility of the protein. In the third step, a feature vector for each generated map is computed, resulting in three feature vectors for each protein. They can be concatenated to a vector containing two or three of the original vectors. We tested three different methods to compute these features: Image Moments [Hu62; Flu00], Color Moments [MSM09], and a Convolutional Neural Network (CNN) [SHZ\*18]. For the Color Moments, we use a color mapping from the scalar values to RGB values (see Figure 3). The other two methods work directly on the scalar maps. In the fourth step, the distances between the feature vectors are used to cluster the proteins hierarchically.

##### 4.1. Protein Alignment

The visual appearance of the Molecular Surface Maps highly depends on the orientation of the protein. Even for very similar proteins, however, the orientation can differ widely in the Protein



Data Bank [BWF\*00]. Therefore, it is advisable to align the proteins as good as possible prior to the mapping. For similar proteins, minimizing the Root Mean Square Deviation (RMSD) between corresponding atoms is the most commonly used alignment method [KS83]. However, since our input data can be highly heterogeneous, this approach is not feasible, as it is not possible to establish the necessary per-atom correspondence for highly different proteins. Thus, we use a more general approach based on Principle Component Analysis of the atom positions: we rotate each protein so that the first principle component is aligned to the x-axis and the second principle component is aligned to the y-axis. Thus, proteins with similar shape will be oriented similarly in a fast and convenient way, regardless of the underlying chemical sequence. This works well for elongated as well as globular proteins as possible flipping in either x- or y-direction is later handled by the rotationally invariant feature extraction methods.

#### 4.2. Molecular Surface Map Creation

The next step after the alignment is to create the three maps for each protein using the Molecular Surface Map algorithm proposed by Krone et al. [KSR\*17], which is shown schematically in Figure 2. Their algorithm computes the SES for a given probe radius and ensures that the SES is of genus zero by detecting and closing tunnels through the protein. The next step is the computation of a transformation from the SES to a sphere. Finally, the two-dimensional map is created by using a suitable map projection, e.g., Lambert equal area projection, which minimizes the area distortion at the poles. We changed their algorithm so that the generated map contains the scalar values of the property directly. Additionally, we replaced the costly tunnel detection and closing part of the original algorithm with our own variant, which is fast and, thus, results in lower computation times for large protein ensembles. Our approach iteratively increases the probe radius, used to compute the SES mesh, from 2.4 Å, which was proposed by domain experts in biochemistry, to 4.0 Å or until the mesh is of genus zero, in steps of 0.2 Å. For cases where our faster approach fails, the original, slower variant by Krone et al. has to be used. However, in our tests, this happened only for less than 10% of the proteins. Our approach can lead to the comparison of surfaces generated for different probe radii. As a change of the probe radius only affects small local concave parts of the surface the effect on the resulting surface are minimal.

#### 4.3. Feature Computation

In order to cluster the Molecular Surface Map images generated in the previous step, we assign a descriptive value to each of them. We tested three approaches: Image Moments, Color Moments, and a feature computation based on a Convolutional Neural Network (CNN). Each method calculates a feature vector—with seven, nine, and 1792 elements respectively—in order to uniquely represent the image. In Section 6, we present an evaluation of the different feature vector calculations. For Image Moments and the CNN features, we directly employ the calculated maps as input. To be able to properly use Color Moments, a color map has to be applied to the scalar maps first. We are aware that the quality of the result heavily depends on the quality of the used color map, but otherwise, the feature vector would not be long enough to accurately represent the

image. Using either of these approaches, each Molecular Surface Map computed in the first step is assigned a feature vector. To create a more unique and meaningful feature vector, multiple Molecular Surface Maps representing different quantities of the same protein can be combined by simply concatenating the feature vectors. Concatenating the feature vectors can for example take mutations of the proteins into account that only change certain quantities. In the following, we will briefly describe the three methods to calculate the feature vectors.

**Image Moments** Image Moments are derived from the intensity value of a pixel  $I(x,y)$ . Therefore, we are able to use our previously calculated Molecular Surface Maps directly. In order to get invariance under translation, rotation and scaling we computed the Hu moments invariants [Hu62]. We left out  $I_3$  since it is dependent on the other moments and computed  $I_8$  instead, as proposed by Flusser [Flu00]. This results in a feature vector  $F_{im}$  for each Molecular Surface Map containing seven moments:

$$F_{im} = (I_1, I_2, I_4, I_5, I_6, I_7, I_8) \quad (1)$$

**Color Moments** We compute the Color Moments as proposed in the work of Maheshwari et al. [MSM09] using the color-coded, and not the scalar value, map as input (see Figure 3 top row). The first three elements in the resulting feature vector are the mean values of each RGB color channel. The next three components of the feature vector are the standard deviation, again for each color channel. The final three components represents the skewness for each color channel separately. This can be understood as a measure of the degree of asymmetry in the distribution. The combined feature vector  $F_{cm}$  for the Color Moments contains nine elements and uniquely describes the associated map.

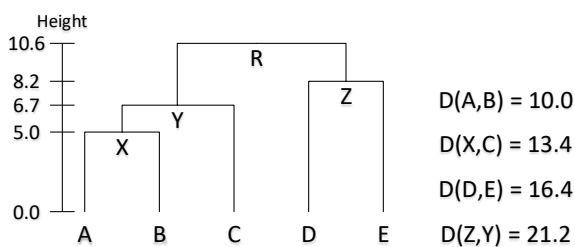
$$F_{cm} = (E_R, E_G, E_B, SD_R, SD_G, SD_B, S_R, S_G, S_B) \quad (2)$$

**Convolutional Neural Network** The third feature computation method we apply is a CNN called MobileNetV2, which was developed by Sandler et al. [SHZ\*18] and is provided as a TensorFlow [AAB\*15] module. We do not apply the whole network to our images, since the final output is not a feature vector. Due to its modular structure, the authors provide a smaller version of the network that only calculates image feature vectors, which we can use as input features for our subsequent calculations. Compared to the other two feature extraction methods, the feature vectors retrieved from MobileNetV2 are significantly longer. For each map, we obtain a feature vector  $F_{nn}$  with 1792 elements, which is invariant under translation, rotation and scaling:

$$F_{nn} = (F_0, F_1, \dots, F_{1791}) \quad (3)$$

#### 4.4. Hierarchical Clustering

In this step of our algorithmic pipeline, a hierarchical clustering of the  $n$  input proteins based on the feature vectors described in Subsection 4.3 is computed. We calculate the Euclidean distance  $d_{ij}$  between each pair of feature vectors in order to get the  $n \times n$  distance matrix. The resulting distance matrix is then used in the iterative process to compute a binary tree containing all maps or proteins respectively. For each of the proteins, a leaf node is created containing the feature vector and the corresponding Molecular



**Figure 4:** UPGMA algorithm [SM58] for five nodes, A to E. The complete distance matrices of each iteration are not shown, but the shortest distances of each iteration are shown on the right side of the tree. The height of the connection is computed as  $h_0 = D_0/2$  for the first level and  $h_i = D_i/2 - D_{i-1}$  for the other levels.

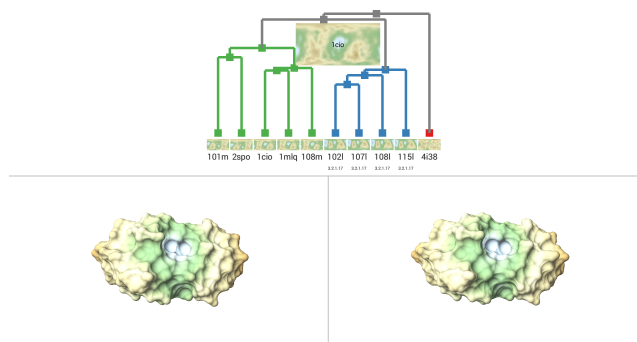
Surface Map. Initially, each leaf node is regarded as a one-element cluster. The further construction of the tree follows the Unweighted Pair Group with Arithmetic Mean (UPGMA) algorithm developed by Sokal and Michener [SM58]. In bioinformatics, UPGMA is widely used to construct phylogenetic trees. Their algorithm follows a bottom-up approach by subsequently merging the clusters with the shortest pair-wise distance. After each merge, the distance matrix is updated by removing the two merged clusters and adding the newly created cluster instead. New distance values from the new cluster  $C_N$  to all existing ones  $C_i$  are calculated as follows:

$$d(C_N, C_i) = \frac{1}{|C_N| \cdot |C_i|} \sum_{x \in C_N} \sum_{y \in C_i} d(x, y) \quad (4)$$

Therefore, the new distance value  $d(C_N, C_i)$  is equal to the arithmetic mean of all pairwise node distances. The new cluster is assigned a descriptive feature vector as well, which is the centroid computed from the feature vectors of all leaf nodes contained in the two merged subtrees. Based on this centroid, we search for the leaf node in the subtree of  $C_N$  with the most similar feature vector and use its map as the representative map of the cluster. After the iterations finishes we have a binary tree containing all maps that provides a preliminary hierarchical clustering. The UPGMA algorithm is able to generate trees where the distance in the tree following the edges corresponds the actual distance in the distance matrix (see Figure 4). This is achieved by assigning a height value that is equal to half the distance of the sub-trees to each newly created cluster node. Constructing the tree in this way allows for an intuitive visual analysis, as the distance values can be estimated directly from the visualization. Using a user-defined similarity threshold  $T_c$  we determine the final clusters based on the UPGMA tree. All nodes that belong to a subtree of less than height  $T_c$  belong to the same cluster (dashed orange line in Figure 1).

#### 4.5. Visualization Application

In order to visualize and analyze the aforementioned clustering, we provide the user with three linked views that can be used to interactively explore and analyze the generated hierarchical clustering (see Figure 5). It is possible to adjust the size of all views by moving the gray lines separating them. In the top view, the computed hierarchical tree is shown using color coding to highlight which subtree



**Figure 5:** Overview over the interface used to display the result of the clustering. The top view shows the hierarchical binary tree, each node colored according to the cluster it belongs. In the two bottom views the SES of selected maps can be shown for further analysis. Here, 102L (left) and 107L (right) are selected. The clustering was performed on the heatmap coloring as this produces results that can be easily confirmed visually by the user.

corresponds to which cluster. The leaf nodes of the tree are the proteins used in the clustering. Hovering over a node in the tree shows the map of the protein associated with that node. For nodes further up in the tree, the most representative map of the subtree is shown. The two remaining views, bottom left and bottom right, are used to show the three-dimensional surface representation of two proteins that were selected in the tree for further analysis. Zooming is possible inside all views, while only the bottom two views support 3D interactions like translations and rotations. The 3D views can be coordinated views, that is, the aligned proteins will rotate and translate simultaneously. The initial state, after the data is loaded, is to show the root node of the hierarchical tree and the bottom views are empty.

#### 5. Description of Methods used for Verification

As mentioned above, there is a wide variety of methods to compare and cluster proteins (see Section 2 and Section 3). In order to verify that the results of our proposed hierarchical clustering method are valid, we compare them with two other resources: the BRENDA data base that provides information about the function of enzymes (Enzyme Classification, EC), and the tool MM-align that uses the TM-score to rate the similarity of two proteins. In this section, we briefly describe these two methods. The results of the verification are discussed in Section 6.

**Enzyme Classification** To test whether our approach is able to find clusters of proteins with similar function, real-world pre-clustered results are necessary. Therefore, we used the enzyme database BRENDA (BRAunschweig ENzyme DATAbase [SCS02]) to retrieve such information. The BRENDA database groups the entries based on their function, following the Enzyme Classification (EC), a naming and numbering convention [Web92]. Each stored protein is assigned a hierarchical EC code number consisting of four digits. For example, the *hydroxynitrile lyase* from almond

**Table 1:** Assigned distance values when comparing two enzyme classifications of the form EC 1.2.3.4. The column headers represent their position in the hierarchical order. The table lists a ✓ when the class on that position matches and a × otherwise.

1	2	3	4	assigned distance
×	× ✓	× ✓	× ✓	4
✓	×	×	× ✓	3
✓	✓	×	× ✓	2
✓	×	✓	× ✓	2
✓	✓	✓	×	1
✓	✓	✓	✓	0

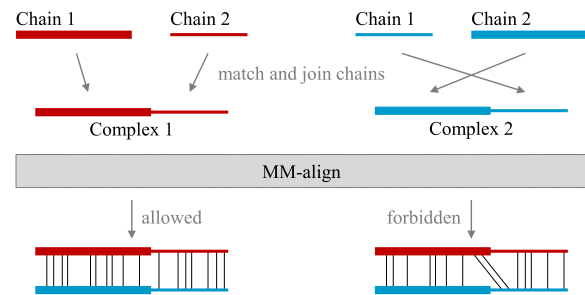
(PDB ID: 1JU2) belongs to the group EC 4.1.2.10. The first number indicates the general role of the enzyme, such as the type of reaction that it is able to accelerate. The meaning of the second two numbers depends on the first one. One number, for example, could indicate the donor of the reaction, another the acceptor. A general description of the last number is not available, as it just allows to further subdivide the pre-defined classes. Therefore, this classification does not represent a true hierarchical clustering, as the order of the inner values is just defined as-is, and could be interchangeable. To turn a comparison between two enzyme classes into a number indicating the quality of the comparison, we assign distance values to specific configurations, which are shown in Table 1. Please note that the exact scalar distance values are arbitrary as we try to convert a nominal categorization into a linear one.

**MM-align & TM-score** Another widely used (e.g., [FT20; GKJ19]) protein similarity measurement score is the *TM-score* (Template Modeling score) [ZS04]. It compares the amino acid sequence and 3D structural information of two proteins and returns a single similarity value, which can be seen as a global fold similarity score. Using MM-align (MultiMer-Align) [MZ09] this score, for pairs of single protein chains, can be computed. In general, MM-align reorders the chains of one of the proteins in order to find an optimal alignment between corresponding amino acids in the two protein complexes. Based on this sequence alignment, a heuristic, iterative algorithm is used to superimposes the structures.

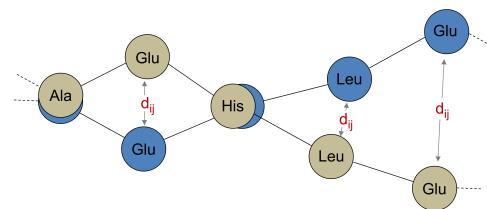
For the reordering, MM-align first permutes all chains of the protein with more or equal chains ( $n \geq m$ ). For each permutation, the sequences of the chains are joined to a new contiguous sequence (see Figure 6). Then, five different sequence alignments are performed on these contiguous sequences (for details about the alignment algorithm, please refer to the original publications). For each of these alignments, an inter-complex distance matrix is computed and used to guide a heuristic, iterative superposition algorithm, which tries to optimize the TM-score (see Figure 7). The iteration stops when the alignments and, consequently, the TM-score converges to a stable number. The TM-score is computed as follows:

$$\text{TM-score} = \max \left[ \frac{1}{L} \sum_{i=1}^{L_{\text{align}}} \frac{1}{1 + \left( \frac{d_{ij}^2}{d_0^2} \right)} \right]$$

where  $L$  denotes the length of all chains of the target complex



**Figure 6:** Example of chain matching and joining procedure and the alignment principle of MM-Align. Two chains (thick and thin) of two different dimeric protein complexes (red, blue) are joined to a contiguous artificial chain (a complex) for each protein. Then these complexes are aligned. MM-align allows only the alignment of amino acids between one chain of the complex to another chain of the other complex (bottom left). Aligning amino acids of one chain of a complex to more than one chain in the other complex is forbidden (bottom right). (Figure modified from [MZ09])



**Figure 7:** Schematic of a superposition between two short parts of aligned amino acid chains (blue and olive circles; Ala = Alanin, Glu = Glutamin, His = Histidin, Leu = Leucin).  $d_{ij}$  denotes the distance between the  $\alpha$ -carbon atoms of two aligned amino acids  $i$  and  $j$ .

and  $L_{\text{align}}$  is the number of aligned protein pairs.  $d_{ij}$  is the Euclidean distance between the  $C\alpha$  atoms of the aligned amino acids  $i$  and  $j$  (cf. Figure 6) and  $d_0$  is a scale to normalize the match difference [ZS04; MZ09]:  $d_0 = 1.24\sqrt[3]{L} - 15 - 1.8$ . Thus, the final TM-score combines the similarity of aligned regions and the alignment coverage as one similarity value. Its value range is  $[0 \dots 1]$ , where higher values denote higher similarity. Additionally, a TM-score above 0.5 indicates similar topology, chain orientation, and the same fold based on SCOP/CATH [MBHC95; OMJ\*97], which makes it useful for protein classification problems [XZ10].

## 6. Results

The goal of our presented clustering pipeline is to find proteins that not only have similar structure but also similar function. We tested our approach using three protein ensembles obtained from the Protein Data Bank [BWF\*00]. The first ensemble consists of ten selected proteins with previously known properties. The second one consist of 50 proteins, the third one of 1000. We verify the results using the EC code provided by BRENDA and the TM-score as mentioned in Section 5.

**Table 2:** Colour coded results of the clustering for the small dataset using each of the three feature extraction methods: Image Moments (IM), Color Moments (CM) and MobileNetV2 (MN) and different maps: Heightmap (H), Hydrophobicity (Y), B-factor (B) and the combinations H-Y and H-Y-B. The expected result is two clusters, C1 and C2, as well as one outlier O. Configurations marked blue resulted in a proper clustering, red configurations were not successful. Yellow values mark borderline cases, where a wrong protein was added to a cluster at a late point in time, indicating large distance to the rest of the cluster. For H-Y and H-Y-B we combine the feature vectors of the maps above either without or with the B-factor features, respectively.

Map	IM			CM			MN		
	C1	C2	O	C1	C2	O	C1	C2	O
H	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue
Y	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue
B	Red	Red	Red	Red	Red	Red	Red	Red	Red
H-Y	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue
H-Y-B	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue	Blue

We constructed the first and smallest data set of ten proteins, so that it would contain two clusters and one outlier. This was achieved by selecting three sufficiently different proteins. After that, a sequence similarity search was performed on two of the three proteins. Out of these search results we picked the most similar proteins to be included into the cluster. The results are shown in Figure 8. We tested all three feature extraction methods using different Molecular Surface Maps and combinations, as shown in Table 2. The correct result for this data set, considering the EC code provided by BRENDA as ground truth, is a clustering where the four enzymes form one cluster, and five of the six remaining proteins are also clustered together, since they are nearly identical. Using MobileNetV2, the two clusters and the outlier are generally well preserved (Figure 8a and Figure 8b). The only exception is the b-factor map in Figure 8c, where only one cluster is preserved properly. This is also true for the other feature extraction methods, Image and Color Moments. When comparing the resulting clustering trees of the different methods shown in Figure 8, one can notice that Image Moments and Color Moments tend to have a more diverse distance value distribution than the CNN features. That means that the distance between the closest proteins is far smaller compared to the distance between the farthest ones. Therefore, many merges happen at the very bottom of the clustering tree as opposed to the MobileNetV2 results. We attribute this to the length of the feature vectors. As the feature vector given by MobileNetV2 has 1792 entries, even small distances for single vector elements add up to larger numbers. Although the base difference values are typically high, the quality of the results is superior to the other two methods.

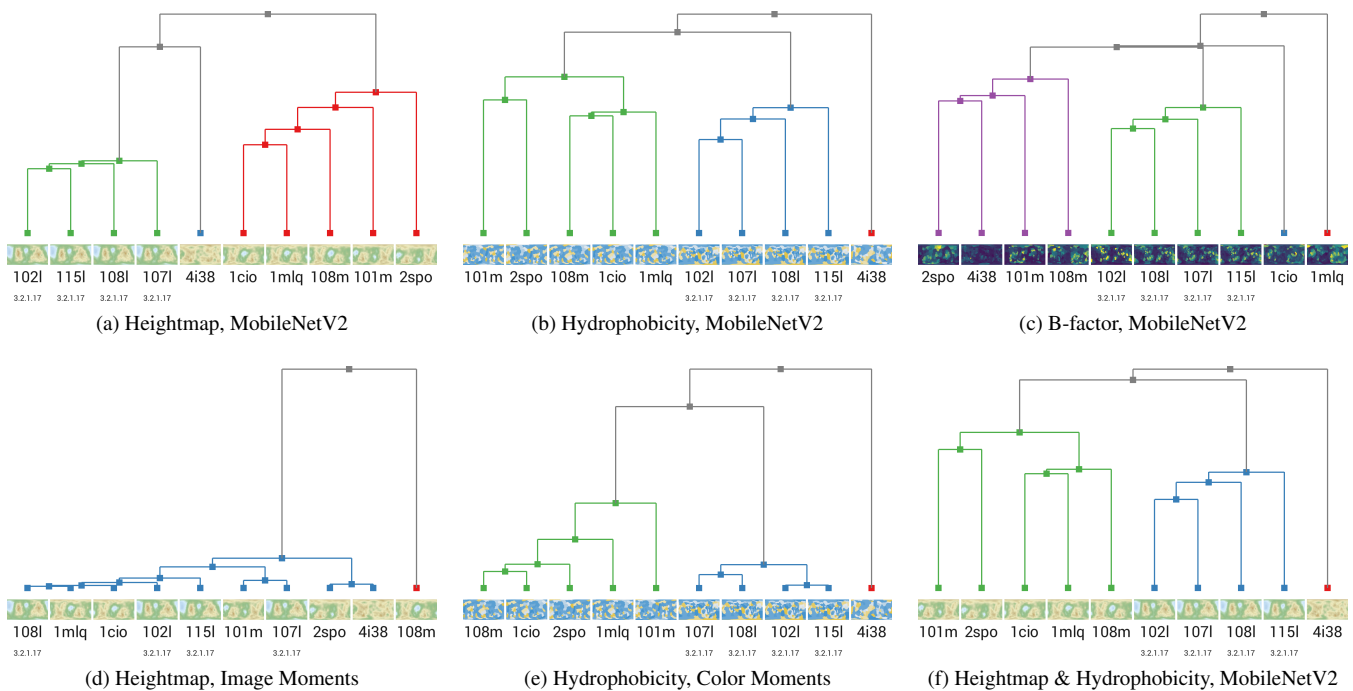
We originally chose the three presented map types to represent different properties of the proteins: the heightmap represents the geometry of the surface, the hydrophobicity can affect binding behavior, and the b-factor is an indicator for flexibility. Initially we considered the b-factor as an interesting feature for the function of a protein, as it can influence the catalytic rate. However, as it is not decisive for the function and it can change under different envi-

ronmental conditions, we decided to omit it for the larger data sets due to the poor results for the small ensemble. We discussed this choice with a biochemist, who confirmed our assessment that the b-factor (or other flexibility measures) are not decisive for a large-scale comparison of different proteins. Only in specific cases, such as when analyzing an ensemble of similar proteins (e.g., different mutants of an enzyme), a clustering based on flexibility could be interesting. This is, however, not the focus of our current work. For the other two map types, the clustering quality depends on the used feature extraction method. Image Moments were successfully used by Kolesár et al. [KBP\*16] to calculate similarities for maps of protein cavities. In our use case, however, they performed worse than Color Moments and MobileNetV2. They even considered one of the enzymes as a clear outlier while merging the actual outlier early. We assume that Kolesár et al. obtained good results since their maps consisted of large distinct patches that differed only slightly between the maps. Although Color Moments strongly depend on the chosen color map, they performed considerably better than Image Moments. However, they exhibit weaknesses in the detection of outliers (cf. Table 2). MobileNetV2 produced the overall best results, especially when considering that the results of using Color Moments features might get worse when choosing a different color map. Additionally, it was the only method that got the correct result for the combined feature vectors.

As mentioned in Section 4, our algorithm can also use a combination of multiple maps for each protein for the clustering. This can be useful since surface properties like the hydrophobicity as well as the shape of the surface influence the function of a protein. As shown in Table 2 and Figure 8f the MobileNetV2-based clustering produces very good results. Since MobileNetV2 performed best on our small verification data set, we only used this method and the combination of heightmap and hydrophobicity map for the two larger data sets. By default, the heightmaps are displayed in our prototype, as they are more easily distinguishable for a human viewer, but users can also choose to see the hydrophobicity map.

Figure 9 shows results for the ensemble of 50 proteins. When investigating the clustered proteins, it can be observed that visually as well as functionally similar enzymes are linked together early, e.g., as seen in the purple boxes. These proteins which were identified as being similar by our method are also classified as being functionally similar according to the EC classification stored in BRENDA. Note that we only color the branches of direct merges of leaf nodes, as those are most easily interpretable and it is not directly clear which value to assume as basis for coloring for the other cases. Overall, the majority of our clustering results for enzymes are consistent with the EC classification. That is, proteins of the same class, and therefore, same function are in the same cluster. Two outlier pairs are visible, IAGX and IAKL, and IAF0 and IAI2. The first pair actually belongs to the same main class and their shape is at least roughly similar, which explains the clustering by our method. That is, the remaining pair the only real outlier, where their heightmap are different and they are not in the same EC class. The clusters in the blue and red boxes are especially interesting when we compare these against the TM-score values. Here, the TM-score indicates a high dissimilarity between these proteins although they are functionally very similar according to their EC classification. With our method, these cases are clustered





**Figure 8:** Comparison of several similarity trees of our given ten protein data set. The subcaptions list the coloring mode alongside with the used feature extraction method. All subfigures display, from top to bottom, the generated tree, the maps belonging to each node, their respective PDB identifier, and, if available, their enzyme classification. Proteins with a displayed enzyme classification should form one cluster, the remaining proteins another one. 4I38 is considered an outlier. Using MobileNetV2 features our method generates the expected result fully ((a) and (b)) or at least partially (c). In contrast, Image Moments shown in (d) completely fails to categorize the proteins similarity properly. Color Moments (e) leads to similarly good results. Finally, (f) shows the combination of two map types that mutually alleviate the shortcomings of each single map type.

early, but typically later than cases where all four enzyme classes match. Additionally, the TM-score was not able to catch the similarity of 1A7T and 1A8T completely, although they fall into the same class (right half of red box). Although the heightmaps do not look very similar, our method was able to correctly cluster them. Failed similarity detections by the TM-score can happen where the interior of two proteins is different but the surfaces are similar. Our method will then detect a high similarity while the TM-score indicates a high difference. As mentioned in Section 2, the surface of a protein is generally more important to its function, that is, matching proteins with similar surface but dissimilar interior is desirable. Thus, we rate the results of our method as highly suitable overall.

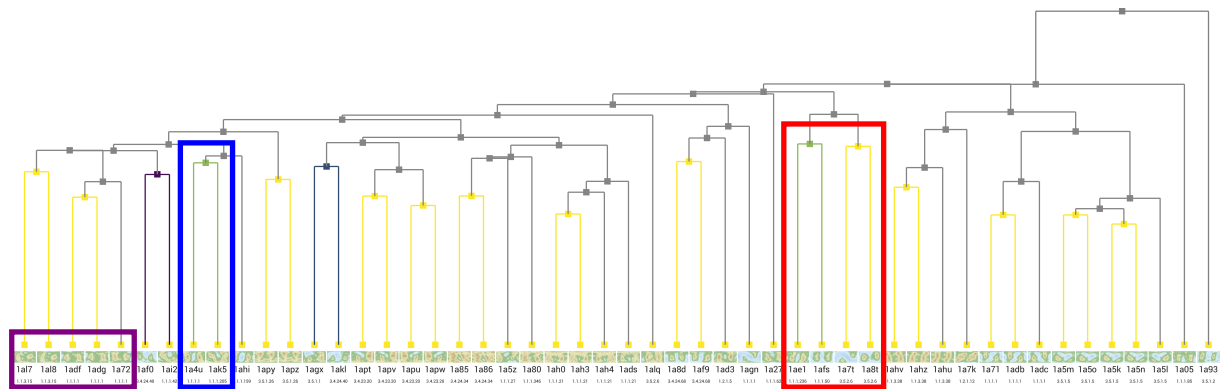
The largest ensemble we tested consists of 1000 proteins, which are all listed in the BRENDA database. The resulting graph is far too large to show every detail without being able to interact with the visualization (panning and zooming). The trends already detected in the 50 protein ensemble can also be found here: a majority of the clusters are of good quality, while the results that contradict the enzyme classification usually happen higher up. Legibility is a general issue of the explicit visualization of larger hierarchical data sets [SHS11], and could be resolved by exploiting implicit methods like tree maps. However, using these methods would imply to give up the explicit representation that domain scientists are

already used to. Using a high resolution display—e.g., a power-wall [MRE13]—could improve the readability of the graph.

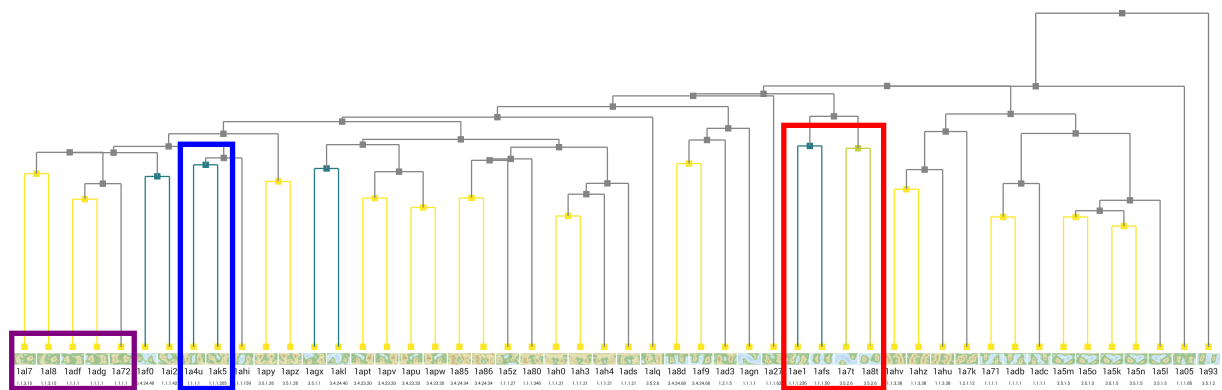
We have shown that the results of our method are overall in good agreement with the enzyme classification and the TM-score. That is, our method can be applied for clustering functionally similar proteins. It produces only few outliers compared to the EC classification obtained from BRENDA that we use as a ground truth. However, other methods for predicting protein-protein similarity like MM-align are also not flawless when it comes to the detection of functional similarity. Our prototypical visualization tool can be used to interactively explore and analyze the hierarchical clustering. This allows expert users to investigate the results in the dendrogram and interactively set the threshold determining the clusters. A detailed analysis of individual pairs of proteins is possible in the 3D views (see Figure 5), which helps to assess the quality of our map-based similarity score. Consequently, our application can be used to cluster and visually analyze previously unknown proteins and hypothesize about the function of the protein.

## 7. Summary and Future Work

We presented an approach to hierarchically cluster and analyze protein ensembles based on surface similarity. To this end, we adapted the Molecular Surface Map method [KSR\*17] and used the result-



(a) Direct merges colored by enzyme classification



(b) Direct merges colored by TM-score

**Figure 9:** Clustering of 50 different proteins using MobileNetV2 feature vectors for heightmap and hydrophobicity. Visually similar maps are clustered together. Merges of two leaf nodes are colored by the quality of the merge based on two different external scores or classifications. Direct matches are colored using the Viridis color map (purple to yellow), where yellow corresponds to a good match and purple to a poor one. (a) uses a given enzyme classification as ground truth, (b) the TM-score. The purple box contains five enzymes belonging to the same class, where the TM-score as well as our method detect a high similarity. The red and blue boxes are cases where our method gives a more suitable similarity score than the TM-score.

ing maps that represent different properties of the protein surface to extract feature vectors, using a Convolutional Neural Network. They are used to construct a dendrogram representing the clustering hierarchy. To test the quality and validity of our results, we investigated whether proteins that were classified as similar by our method also exhibit a functional similarity. Additionally, we compared our results to another similarity scoring method, the TM-Score [ZS04]. As we are intentionally only comparing the surface properties of the proteins while the TM-Score also incorporates the interior, our method results in better classifications in some cases.

In the future, we plan to evaluate the performance of a neural network that is trained specifically for the feature extraction. This could improve the quality of the clustering further since the CNN we used was originally trained for different data but performs very well. Additionally, we plan the exploration of additional distance measures apart from the Euclidean distance, as well as an evaluation on how well a adapted TM-score that weighs surface-facing proteins higher compares with our method. Alternatively, we plan to evaluate statistical methods to compare images have been de-

veloped. For example, the Earth Mover's Distance [RTG00] or the Bhattacharyya distance [CA79] can be exploited for such comparisons. Furthermore, we want to adapt our visualization prototype so that it runs on large high-resolution displays. The larger screen space could help with the visual analysis of the results, since exploring the dendrogram for larger ensembles of proteins is a tedious task on regular displays.

#### Acknowledgments

This work was partially funded by German Research Foundation (DFG) as part of SFB-TRR 161 (project ID 251654672), as well as part of project PROLINT (project ID 391088465), and by Carl-Zeiss-Stiftung.

#### References

[AAB\*15] ABADI, MARTÍN, AGARWAL, ASHISH, BARHAM, PAUL, et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.*

- Software available from tensorflow.org. 2015. URL: <https://www.tensorflow.org/5>.
- [ABK\*96] ANZALI, SOHEILA, BARNICKEL, GERHARD, KRUG, MICHAEL, et al. "The comparison of geometric and electronic properties of molecular surfaces by neural networks: Application to the analysis of corticosteroid-binding globulin activity of steroids". *J. Comput. Aid. Mol. Des.* 10.6 (1996), 521–534 3.
- [AGM\*90] ALTSCHUL, STEPHEN F, GISH, WARREN, MILLER, WEBB, et al. "Basic Local Alignment Search Tool". *J. Mol. Biol.* 215.3 (1990), 403–410 2.
- [BGG07] BOCK, MARY ELLEN, GARUTTI, CLAUDIO, and GUERRA, CONCETTINA. "Discovery of Similar Regions on Protein Surfaces". *J. Comput. Biol.* 14.3 (2007), 285–299 3.
- [BJ09] BURGOYNE, NICHOLAS J. and JACKSON, RICHARD M. "Predicting Protein Function from Surface Properties". *From Protein Structure to Function with Bioinformatics*. Ed. by RIGDEN, DANIEL JOHN. Springer Netherlands, 2009, 167–186. DOI: [10.1007/978-1-4020-9058-5\\_7\\_2](https://doi.org/10.1007/978-1-4020-9058-5_7_2).
- [BTSC02] BERG, JEREMY M., TYMOCZKO, JOHN L., STRYER, LUBERT, and CLARKE, NEIL D. *Biochemistry*. 5. ed., 4. print. New York, NY: W. H. Freeman, 2002. ISBN: 0-7167-3051-0 2.
- [BvLH\*11] BREMM, SEBASTIAN, von LANDESBERGER, TATIANA, HESS, MARTIN, et al. "Interactive visual comparison of multiple trees". *IEEE Conference on Visual Analytics Science and Technology (VAST)*. 2011, 31–40 4.
- [BWF\*00] BERMAN, HELEN M., WESTBROOK, JOHN, FENG, ZUKANG, et al. "The Protein Data Bank". *Nucleic Acids Research* 28.1 (2000), 235–242. DOI: [10.1093/nar/28.1.235\\_5\\_7](https://doi.org/10.1093/nar/28.1.235_5_7).
- [CA79] COLEMAN, GUY BARRETT and ANDREWS, HARRY C. "Image segmentation by clustering". *Proceedings of the IEEE* 67.5 (1979), 773–785 10.
- [CHL\*04] CAI, DENG, HE, XIAOFEI, LI, ZHIWEI, et al. "Hierarchical clustering of WWW image search results using visual, textual and link information". *ACM International Conference on Multimedia*. 2004, 952–959. DOI: [10.1145/1027527.1027747\\_4](https://doi.org/10.1145/1027527.1027747_4).
- [Con83] CONNOLLY, MICHAEL L. "Analytical molecular surface calculation". *J. Appl. Cryst* 16.5 (1983), 548–558 3.
- [Flu00] FLUSSER, JAN. "On the independence of rotation moment invariants". *Pattern Recognition* 33.9 (2000), 1405–1410 2, 4, 5.
- [FM67] FITCH, WALTER M and MARGOLIASH, EMANUEL. "Construction of Phylogenetic Trees". *Science* 155.3760 (1967), 279–284 2, 4.
- [FT20] FUKUDA, HIROYUKI and TOMII, KENTARO. "DeepECA: an end-to-end learning framework for protein contact prediction from a multiple sequence alignment". *BMC Bioinformatics* 21.1 (2020), 10. DOI: [10.1186/s12859-019-3190-x\\_7](https://doi.org/10.1186/s12859-019-3190-x_7).
- [GKJ19] GREENER, JOE G., KANDATHIL, SHAUN M., and JONES, DAVID T. "Deep learning extends de novo protein modelling coverage of genomes using iteratively predicted structural constraints". *Nature communications* 10.1 (2019), 3977. DOI: [10.1038/s41467-019-11994-0\\_7](https://doi.org/10.1038/s41467-019-11994-0_7).
- [HF12] HASEGAWA, KIYOSHI and FUNATSU, KIMITO. "New description of protein–ligand interactions using a spherical self-organizing map". *Bioorg. Med. Chem.* 20.18 (2012), 5410–5415 3.
- [HK14] HASS, JOEL and KOEHL, PATRICE. "How round is a protein? Exploring protein structures for globularity using conformal mapping". *Front. Biosci.* 1 (2014), 26. DOI: [10.3389/fmolsb.2014.00026\\_3](https://doi.org/10.3389/fmolsb.2014.00026_3).
- [HLA04] HOFBAUER, CHRISTIAN, LOHNINGER, HANS, and ASZÓDI, ANDRÁS. "SURFCOMP: A Novel Graph-Based Approach to Molecular Surface Comparison". *J. Chem. Inf. Comput. Sci* 44.3 (2004), 837–847 4.
- [HRR\*07] HUSON, DANIEL H, RICHTER, DANIEL C, RAUSCH, CHRISTIAN, et al. "Dendroscope: An interactive viewer for large phylogenetic trees". *BMC Bioinformatics* 8.1 (2007), 460 4.
- [Hu62] HU, MING-KUEI. "Visual pattern recognition by moment invariants". *IRE Trans. Inf. Theory* 8.2 (1962), 179–187 2, 4, 5.
- [Hun07] HUNTER, JOHN D. "Matplotlib: A 2D Graphics Environment". *Computing in Science Engineering* 9.3 (2007), 90–95. ISSN: 1558-366X. DOI: [10.1109/MCSE.2007.55\\_3](https://doi.org/10.1109/MCSE.2007.55_3).
- [KBP\*16] KOLESÁR, IVAN, BYŠKA, JAN, PARULEK, JULIUS, et al. "Unfolding and Interactive Exploration of Protein Tunnels and their Dynamics". *Eurographics Workshop on Visual Computing for Biology and Medicine*. 2016. DOI: [10.2312/vcbm.20161265\\_3\\_8](https://doi.org/10.2312/vcbm.20161265_3_8).
- [KGE11] KRONE, MICHAEL, GROTTTEL, SEBASTIAN, and ERTL, THOMAS. "Parallel contour-buildup algorithm for the molecular surface". *IEEE Symposium on Biological Data Visualization (BioVis)*. 2011, 17–22 3.
- [KKF\*17] KOZLÍKOVÁ, BARBORA, KRONE, MICHAEL, FALK, MARTIN, et al. "Visualization of Biomolecular Structures: State of the Art Revisited". *Comput. Graph. Forum* 36.8 (2017), 178–204. DOI: [10.1111/cgfm.13072\\_2\\_3](https://doi.org/10.1111/cgfm.13072_2_3).
- [Koe01] KOEHL, PATRICE. "Protein structure similarities". *Curr. Opin. Struct. Biol.* 11.3 (2001), 348–353. DOI: [10.1016/S0959-440X\(00\)00214-1\\_2](https://doi.org/10.1016/S0959-440X(00)00214-1_2).
- [KS83] KABSCH, WOLFGANG and SANDER, CHRISTIAN. "Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features". *Biopolymers* 22.12 (1983), 2577–2637. DOI: [10.1002/bip.360221211\\_5](https://doi.org/10.1002/bip.360221211_5).
- [KSR\*17] KRONE, MICHAEL, SCHARNOWSKI, FLORIAN FRIESSAND KATRIN, REINA, GUIDO, et al. "Molecular Surface Maps". *IEEE Trans. Vis. Comput. Graph.* 23.1 (2017), 701–710. DOI: [10.1109/TVCG.2016.2598824\\_1-5\\_9](https://doi.org/10.1109/TVCG.2016.2598824_1-5_9).
- [KVTK16] KONTOPOULOS, DIMITRIOS GEORGIOS, VLACHAKIS, DIMITRIOS, TSILIKI, GEORGIA, and KOSSIDA, SOFIA. "Structuprint: a scalable and extensible tool for two-dimensional representation of protein surfaces". *BMC Struct. Biol.* 16.1 (2016), 4. DOI: [10.1186/s12900-016-0055-7\\_3](https://doi.org/10.1186/s12900-016-0055-7_3).
- [KW05] KOCH, MARCUS A and WALDMANN, HERBERT. "Protein structure similarity clustering and natural product structure as guiding principles in drug discovery". *Drug Discov. Today* 10.7 (2005), 471–483. DOI: [10.1016/S1359-6446\(05\)03419-7\\_2](https://doi.org/10.1016/S1359-6446(05)03419-7_2).
- [LBB\*07] LARKIN, MARK A, BLACKSHIELDS, GORDON, BROWN, NP, et al. "Clustal W and Clustal X Version 2.0". *Bioinformatics* 23.21 (2007), 2947–2948 2.
- [LEV\*09] LA, DAVID, ESQUIVEL-RODRÍGUEZ, JUAN, VENKATRAMAN, VISHWESH, et al. "3D-SURFER: software for high-throughput protein surface comparison and analysis". *Bioinformatics* 25.21 (2009), 2843–2844. DOI: [10.1093/bioinformatics/btp542\\_3](https://doi.org/10.1093/bioinformatics/btp542_3).
- [LR71] LEE, BYUNGKOOK and RICHARDS, FREDERIC M. "The interpretation of protein structures: estimation of static accessibility". *J. Mol. Biol.* 55.3 (1971), 379–400 3.
- [MBHC95] MURZIN, A. G., BRENNER, S. E., HUBBARD, T., and CHOTHIA, C. "SCOP: a structural classification of proteins database for the investigation of sequences and structures". *J. Mol. Biol.* 247.4 (1995), 536–540. ISSN: 0022-2836. DOI: [10.1006/jmbi.1995.0159\\_7](https://doi.org/10.1006/jmbi.1995.0159_7).
- [MPJF95] MOULT, JOHN, PEDERSEN, JAN T, JUDSON, RICHARD, and FIDELIS, KRZYSZTOF. "A large-scale experiment to assess protein structure prediction methods". *Proteins: Structure, Function, and Bioinformatics* 23.3 (1995), ii–iv 2.
- [MRE13] MÜLLER, CHRISTOPH, REINA, GUIDO, and ERTL, THOMAS. "The VVand: A Two-Tier System Design for High-Resolution Stereo Rendering". *CHI POWERWALL 2013 Workshop*. 2013 9.
- [MSM09] MAHESHWARI, MANISH, SILAKARI, SANJAY, and MOTWANI, MAHESH. "Image Clustering Using Color and Texture". *First International Conference on Computational Intelligence, Communication Systems and Networks*. 2009, 403–408. DOI: [10.1109/CICSYN.2009.69\\_2\\_4\\_5](https://doi.org/10.1109/CICSYN.2009.69_2_4_5).

- [MZ09] MUKHERJEE, SRAYANTA and ZHANG, YANG. “MM-align: a quick algorithm for aligning multiple-chain protein complex structures using iterative dynamic programming”. *Nucleic Acids Research* 37.11 (2009), e83. DOI: [10.1093/nar/gkp318](https://doi.org/10.1093/nar/gkp318) 7.
- [OMJ\*97] ORENGO, C. A., MICHIE, A. D., JONES, S., et al. “CATH – a hierarchic classification of protein domain structures”. *Structure* 5.8 (1997), 1093–1109. ISSN: 09692126. DOI: [10.1016/S0969-2126\(97\)00260-8](https://doi.org/10.1016/S0969-2126(97)00260-8) 7.
- [PK14] PANDEY, SHREELEKHA and KHANNA, PRITEE. “A hierarchical clustering approach for image datasets”. *International Conference on Industrial and Information Systems (ICIIS)*. 2014, 1–6 4.
- [Ric77] RICHARDS, FREDERIC M. “Areas, volumes, packing, and protein structure”. *Ann. Rev. Biophys. Bioeng.* 6.1 (1977), 151–176. DOI: [10.1146/annurev.bb.06.060177.001055](https://doi.org/10.1146/annurev.bb.06.060177.001055) 3.
- [RS07] RAHI, SAHAND JAMAL and SHARP, KIM. “Mapping Complicated Surfaces onto a Sphere”. *Int. J. Comput. Geom. Appl.* 17.04 (2007), 305–329. DOI: [10.1142/S0218195907002355](https://doi.org/10.1142/S0218195907002355) 3.
- [RTG00] RUBNER, YOSSI, TOMASI, CARLO, and GUIBAS, LEONIDAS J. “The earth mover’s distance as a metric for image retrieval”. *Int. J. Comput. Vis.* 40.2 (2000), 99–121 10.
- [RZK\*19] RAU, TOBIAS, ZAHN, SEBASTIAN, KRONE, MICHAEL, et al. “Interactive CPU-based Ray Tracing of Solvent Excluded Surfaces”. *Eurographics Workshop on Visual Computing for Biology and Medicine*. 2019. DOI: [10.2312/vcbm.20191249](https://doi.org/10.2312/vcbm.20191249) 3.
- [SCS02] SCHOMBURG, IDA, CHANG, ANTJE, and SCHOMBURG, DIETMAR. “BRENDA, enzyme data and metabolic information”. *Nucleic Acids Research* 30.1 (2002), 47–49 6.
- [SHS11] SCHULZ, H., HADLAK, S., and SCHUMANN, H. “The Design Space of Implicit Hierarchy Visualization: A Survey”. *IEEE Trans. Vis. Comput. Graph.* 17.4 (2011), 393–411 9.
- [SHZ\*18] SANDLER, MARK, HOWARD, ANDREW, ZHU, MENGLONG, et al. “MobileNetV2: Inverted Residuals and Linear Bottlenecks”. *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* 2018, 4510–4520 2, 4, 5.
- [SKB\*19] SCHATZ, KARSTEN, KRONE, MICHAEL, BAUER, TABEA L., et al. “Molecular Sombreros: Abstract Visualization of Binding Sites within Proteins”. *Eurographics Workshop on Visual Computing for Biology and Medicine*. 2019. DOI: [10.2312/vcbm.20191248](https://doi.org/10.2312/vcbm.20191248) 3.
- [SKR\*14] SCHARNOWSKI, KATRIN, KRONE, MICHAEL, REINA, GUIDO, et al. “Comparative Visualization of Molecular Surfaces Using Deformable Models”. *Comput. Graph. Forum* 33.3 (2014), 191–200. DOI: [10.1111/cgf.12375](https://doi.org/10.1111/cgf.12375) 4.
- [SLL\*08a] SAEL, LEE, LA, DAVID, LI, BIN, et al. “Rapid comparison of properties on protein surface”. *Proteins Struct. Funct. Bioinforma.* 73.1 (2008), 1–10 4.
- [SLL\*08b] SAEL, LEE, LI, BIN, LA, DAVID, et al. “Fast Protein Tertiary Structure Retrieval Based on Global Surface Shape Similarity”. *Proteins Struct. Funct. Bioinforma.* 72.4 (2008), 1259–1273 3.
- [SM58] SOKAL, ROBERT R and MICHENER, CHARLES D. “A Statistical Method for Evaluating Systematic Relationships”. *Univ. Kans. Sci. Bull.* 2.38 (1958), 1409–1438 1, 4, 6.
- [SN87] SAITOU, NARUYA and NEI, MASATOSHI. “The Neighbor-joining Method: A New Method for Reconstructing Phylogenetic Trees.” *Mol. Biol. Evol.* 4.4 (1987), 406–425 4.
- [SOS96] SANNER, MICHEL F., OLSON, ARTHUR J., and SPEHNER, JEAN-CLAUDE. “Reduced Surface: An Efficient Way to Compute Molecular Surfaces”. *Biopolymers* 38.3 (1996), 305–320. DOI: [10.1002/\(SICI\)1097-0282\(199603\)38:3<305::AID-BIP4>3.0.CO;2-Y](https://doi.org/10.1002/(SICI)1097-0282(199603)38:3<305::AID-BIP4>3.0.CO;2-Y) 3.
- [SPG\*17] SAXENA, AMIT, PRASAD, MUKESH, GUPTA, AKSHANSH, et al. “A review of clustering techniques and developments”. *Neurocomputing* 267 (2017), 664–681 4.
- [Ste56] STEINHAUS, HUGO. “Sur la division des corp matériels en parties”. *Bull. Acad. Polon. Sci* 1.804 (1956), 801 4.
- [TL12] TSENG, YAN YUAN and LI, WEN-HSIUNG. “Classification of protein functional surfaces using structural characteristics”. *PNAS* 109.4 (2012), 1170–1175. ISSN: 0027-8424. DOI: [10.1073/pnas.1119684109](https://doi.org/10.1073/pnas.1119684109) 2.
- [Web92] WEBB, EDWIN, ed. *Enzyme Nomenclature 1992. Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology and the Nomenclature Classification of Enzymes*. Academic Press, 1992. ISBN: 9781483298689 6.
- [XESK14] XIONG, YI, ESQUIVEL-RODRIGUEZ, JUAN, SAEL, LEE, and KIHARA, DAISUKE. “3D-SURFER 2.0: Web Platform for Real-Time Search and Characterization of Protein Surfaces”. *Protein Structure Prediction* (2014), 105–117 2, 3.
- [XZ10] XU, JINRUI and ZHANG, YANG. “How significant is a protein structure similarity with TM-score = 0.5?”. *Bioinformatics* 26.7 (2010), 889–895. DOI: [10.1093/bioinformatics/btq066](https://doi.org/10.1093/bioinformatics/btq066) 7.
- [ZS04] ZHANG, YANG and SKOLNICK, JEFFREY. “Scoring function for automated assessment of protein structure template quality”. *Proteins: Structure, Function, and Bioinformatics* 57.4 (2004), 702–710. DOI: [10.1002/prot.20264](https://doi.org/10.1002/prot.20264) 1, 2, 7, 10.