Similarity Measures for Protein Structures based on Fuzzy Histogram Comparison

Thomas Fober and Eyke Hüllermeier
Department of Mathematics and Computer Sciences
Marburg University, Germany

Draft of a paper published in Proc. WCCI–2010, World Congress on Computational Intelligence, Barcelona, 2010.

Abstract

We propose a method for comparing protein structures or, more specifically, protein binding sites in terms of histogram-based representations. These representation are intended to capture important geometrical and physico-chemical properties. In comparison to hitherto existing approaches in structural bioinformatics, especially graph-based methods and methods from computational geometry, our approach is computationally much more efficient. Moreover, despite its simplicity, first experimental studies suggest that it is able to produce useful measures of similarity.

1 Introduction

Theory formation in the biological sciences in general and molecular biology in particular is largely founded on similarity-based and analogical reasoning principles. Correspondingly, the comparison of two objects, such as proteins, has become a fundamental problem in bioinformatics. For example, *sequence alignment* is nowadays considered as a standard tool for comparing biomolecules on the *sequence level*.

Structural bioinformatics has gained increasing attention in the past ten years, largely due to the advance of structural databases that offer protein structure information in addition to mere sequence information. With the steady improvement of structure prediction methods, the inference of protein function based on structure information becomes more and more important. Owing to the commonly accepted paradigm stating that similar protein function is mirrored by similar structure (but not necessarily similar sequence), the comparison of protein structures is a central task in this regard.

Obviously, the comparison of (one-dimensional) sequences is less difficult than the comparison of (three-dimensional) molecular structures being charac-

terized by different types of properties, such as geometry and physico-chemical properties. Yet, quite a number of methods for structure comparison have already been proposed in the literature.

One class of methods focuses on geometrical aspects and, correspondingly, makes use of tools from computational geometry. As examples of this type of approach, we mention geometric hashing [14] and the method of *labeled point cloud superposition* recently introduced in [4].

Another idea is to use *graphs* as formal models of molecular structures. Here, the focus is more on the physical and chemical properties, which are often modeled as nodes of a graph, while geometrical or topological properties are captured in a more indirect way via the edges of the graph. Roughly speaking, the problem of comparing biomolecular structures is thus reduced to the problem of comparing graphs. Typical examples of this approach include measures based on sub-graph isomorphism [3, 12], graph edit distance [20, 5, 2], and graph kernels [8, 11].

Geometrical and graph-based approaches are appealing for several reasons. In particular, they produce more than a numerical degree of similarity or, equivalently, distance. Usually, they also provide useful extra information explaining this number. The method of *multiple graph alignment* [20], for example, is a graph-based counterpart to classical sequence alignment that yields (hypothetical) one-to-one correspondences between basic structural units, such as amino acids. The price to pay for this extra information, however, is a high computational complexity. In fact, many of the aforementioned methods lead to NP-complete optimization problems and scale very poorly with the size of the structures. This complexity prevents them from being used in large-scale studies, for example a cluster analysis requiring an all-against-all comparison of many structures.

A possible alternative to methods of the above kind is offered by *feature-based* approaches in which an object is first represented in terms of a fixed number of features, in the simplest case a vector of fixed dimensionality. The comparison of objects is thus reduced to the comparison of feature vectors. Since the original object cannot be recovered from a finite number of features, this transformation normally comes with a significant loss of information. Consequently, it is also unclear how well the similarity of the original objects is mirrored by the similarity of their respective feature vectors. On the other hand, this approach has an obvious advantage with regard to complexity, as feature vectors can be compared quite efficiently.

In this paper, we propose a feature-based approach to the comparison of protein structures, with a special emphasis of protein binding sites. More specifically, our idea is to summarize important information about the geometrical and physico-chemical properties of protein binding sites in terms of histograms or, more generally, fuzzy histograms. To a large extent, this idea is motivated by the successful use of similar approaches in the field of image processing, where the distribution of the brightness or the colors of a picture are represented in terms of histograms, too [15, 19]. In this field, surprisingly strong results (e.g., in terms of classification performance) have been obtained on the basis

of this simple representation. The main goal of this paper is to elaborate on the question whether similar results can be achieved in the context of structural bioinformatics, too.

The remainder of the paper is organized as follows. Subsequent to a brief introduction to the modeling of protein binding sites in Section 2, we introduce different types of histogram representation for binding sites in Section III. In Section IV, several distance measures suitable for comparing histograms will be discussed. Experimental results are presented in Section V, and Section 6 concludes the paper.

2 Modelling Protein Binding Sites

In this paper, our special interest concerns the modeling of protein binding sites. More specifically, our work builds upon CavBase [16], a database for the automated detection, extraction, and storing of protein cavities (hypothetical binding sites) from experimentally determined protein structures (available through the PDB). In CavBase, a set of points is used as a first approximation to describe a binding pocket. The database currently contains 248,686 hypothetical binding sites that have been extracted from 61,516 publicly available protein structures using the LIGSITE algorithm [9].

The geometrical arrangement of the pocket and its physicochemical properties are first represented by predefined *pseudocenters* – spatial points that represent the center of a particular property. The type and the spatial position of the centers depend on the amino acids that border the binding pocket and expose their functional groups. They are derived from the protein structure using a set of predefined rules [16]. As possible types for pseudocenters, hydrogen-bond donor, acceptor, mixed donor/acceptor, hydrophobic aliphatic, metal ion, pi (accounts for the ability to form π–π interactions) and aromatic properties are considered.

Pseudocenters can be regarded as a compressed representation of areas on the cavity surface where certain protein-ligand interactions are experienced. Consequently, a set of pseudocenters is an approximate representation of a spatial distribution of physicochemical properties. Protein binding sites in CavBase are characterized by around 180 pseudocenters on average (even though much larger structures do of course exist).

3 Histogram Representations

A histogram h is a partition of a set of observations $\mathcal{O} \subset \mathcal{X}$ into a finite number of discrete units. Formally, h can be represented as a $\mathcal{B} \longrightarrow \mathbb{R}$ mapping, where $\mathcal{B}$ is a finite set of *bins*, and $h(b)$ denotes the number (fraction) of observations falling into bin b. We call a histogram h normalized if $\sum_{b \in \mathcal{B}} h(b) = 1$. Each bin b is associated with a subset $X[b]$ of the domain $\mathcal{X}$, so that $h(b) = |\mathcal{O} \cap X[b]|$

before normalization and

$$h(b) = \frac{|\mathcal{O} \cap X[b]|}{|\mathcal{O}|}$$

in the normalized case. The set of bins is assumed to form a partition of $\mathcal{X}$, i.e., $X[a] \cap X[b] = \emptyset$ for $a \neq b$ and $\bigcup_{b \in \mathcal{B}} X[b] = \mathcal{X}$. The most common example is a partitioning of the reals, in which case the bins b are associated with intervals $X[b] \subset \mathbb{R}$.

To obtain histograms from a protein binding site, we will use two important properties:

- its distribution of pseudocenters, and
- the distribution of distances between pseudocenters,

thereby capturing both, the physico-chemical properties as well as the geometry of the protein binding site.

3.1 Representation I

The simplest way to derive histograms is to consider both distributions, the distribution of pseudocenters and the distribution of distances, separately, resulting in two histograms for each binding site. Thus, in a first step, we represent a protein binding site by two sets of observations, namely the set

$$P = \{p_1, \ldots, p_n\}$$

of pseudocenters and the set

$$D = \{d_{i,j} = |p_i - p_j| \,|\, p_i, p_j \in P\}$$

of all pairwise distances between the pseudocenters in P (thus, $|D| = n \cdot (n + 1)/2$).

To derive the histogram for pseudocenters, we use the bins $\mathcal{B} = \{1, 2, 3, 4, 5, 6, 7\}$. Thus, for each type of pseudocenter, we simply count its (relative) number of occurrences. For pairwise distances between pseudocenters, we use the set of bins $\mathcal{B} = \{1, \ldots, d_{\max}\} \subset \mathbb{N}$, where $d_{\max}$ is an upper bound on the edge length (measured in the unit Å).[1] So, $h(b)$ is the percentage of edges whose length is in $[b - 1, b[$.

3.2 Representation II

Considering distances and pseudocenters separately obviously comes with a loss of information. To avoid this problem, we also try a second, more complex representation that is based on 28 sets of pairwise distances: $D_{i,j}$ is the set of all distances between pseudocenters of type i and j, with $1 \leq i \leq j \leq 7$.

[1] We determine this number in a pre-processing step by taking the smallest lower bound valid for the data set at hand.

Again, to obtain a corresponding histogram $h_{i,j}$, we use $\mathcal{B} = \{1, \ldots, d_{\max}\}$ and let $X[b] = [b-1, b[$. All histograms are normalized so as to give them the same weight (except empty histograms that remain empty).

The resulting histograms are still one-dimensional, however, this type of representation has the advantage of combining information about pseudocenters and distances. The price to pay is a larger number of histograms along with the need of 28 instead of only two comparisons.

3.3 Fuzzy Histograms

Fuzzy histograms have been introduced as an extension of conventional histograms, mainly to avoid some problems caused by crisp interval boundaries. In fact, these boundaries are to some extent arbitrary, and in many cases, a small change of a boundary may produce a significant change of the shape of the histogram. Fuzzy histograms are intended to be more robust in this regard, especially in the presence of noisy data. For our application, this is especially important, since distances between pseudocenters can vary due to measurement errors or biological variation. Moreover, fuzzy histograms have a smooth instead of a discontinuous shape, which is often more convenient. They have already been used successfully in different application fields, e.g., in image retrieval and classification [18, 19].

The basic idea of fuzzy histograms is to replace bins by "fuzzy bins" b characterized by fuzzy subsets $X[b]$ of $\mathcal{X}$. Thus, each element $x \in \mathcal{X}$ belongs to a bin b to the degree $X[b](x) \in [0, 1]$. In our case, we proceed from a uniform fuzzy partition

$$\mathcal{B} = \{\, X[b] \mid b = 1, 2, \ldots, d_{\max} \,\},$$

where each $X[b]$ is a triangular fuzzy set with core $\{b\}$ and support $]b-w, b+w[$, where $w = 3$ in this paper.

The fuzzy histogram itself, h_f, is then defined as a $\mathcal{B} \longrightarrow \mathbb{R}$ mapping in a straightforward way, namely by replacing counts with sigma-counts. Thus,

$$h_f(b) = \sum_{d \in D} X[b](d) \ ,$$

with D the given set of data (observed distances).

4 Distance Measures

Having reduced the representation of a protein structure to a set of histograms, the problem to compare different structures can be solved by defining proper distance measures on histograms. More specifically, consider two structures with type-I histogram representations (g_1, g_2) and (h_1, h_2), respectively. Moreover, suppose that δ_1 is a distance measure suitable for comparing (pseudocenter) histograms g_1 and h_1, and that δ_2 is a distance measure suitable for comparing

(distance) histograms g_2 and h_2. The overall distance between the two structures can then be defined, for example, by

$$\sqrt{\delta_1(g_1, h_1)^2 + \delta_2(g_2, h_2)^2},$$

i.e., by the L_2-norm of the tuple of distances. Similarly, for the type-II representation, a measure of the form

$$\sqrt{\sum_{i=1}^{28} \delta(g_i, h_i)^2} \tag{1}$$

can be used. Here, only a single distance $\delta(\cdot)$ is needed, since all histograms are of the same type.

In the literature, two types of distance measures are distinguished, namely *bin-by-bin* and *cross-bin* measures. The former are rather simple and only compare the values in the same bin. The overall distance between two histograms is then defined by the sum of distances for all bins. Cross-bin measures, on the other hand, also compare values in different bins. In order to aggregate these distances, these measures also require the existence of a *ground distance* on $\mathcal{B}$.

4.1 Bin-by-Bin Measures

In the following, we recall two important bin-by-bin measures suitable for comparing two histograms g and h, both defined on the same set of bins $\mathcal{B}$.

- *Minkowski Distance:* The well-known Minkowski distance (L_p-norm) is defined as

  $$d_M(g, h) = \left(\sum_{b \in \mathcal{B}} |g(b) - h(b)|^p \right)^{\frac{1}{p}}$$

  and requires the specification of the parameter p. In image retrieval, $p = 1$, $p = 2$ or $p = \infty$ are often used, and we will try the same values for our problem of measuring the distance between protein binding sites.

- *Histogram Intersection:* The Minkowski distance gives the same weight to all bins. This can be a disadvantage, especially if the mass of the two histograms is centered only on a (small) subset of the bins. In particular, bins that are empty in both histograms contribute to their similarity. This is questionable, as it means that, in principle, the similarity can be increased by adding additional (empty) bins.

  A measure avoiding this disadvantage is the (generalized) Jaccard measure, that is commonly used for measuring the similarity between fuzzy sets. Its underlying idea is to compare the size of the intersection of the two sets with the size of their union. In terms of a distance, it can be defined as follows:

  $$d_J(g, h) = 1 - \frac{\sum_{b \in \mathcal{B}} \min(g(b), h(b))}{\sum_{b \in \mathcal{B}} \max(g(b), h(b))} \; .$$

Apart from these two, many other measures could of course be used as well, for example the χ^2 statistic, the Kullback-Leibler-divergence, etc. We did not include these alternatives in our study, mainly since they cause some computational problems. In particular, their computation becomes numerically instable for small values $h(b)$ close to 0. In our application, the probability to encounter such values, or even bins that are completely empty, is rather high.

4.2 Cross-Bin Measures

Bin-by-bin measures essentially treat a histogram as a *set* of unrelated bins. Obviously, this comes with a loss of information if, as in the case of distances between pseudocenters, the underlying domain $\mathcal{X}$, on which the bins are defined, is endowed with a metric structure. In this case, it makes sense to consider two bins as neighbored, or to say that bin a is closer to bin b than to c. Cross-bin measures are able to take such relationships between bins into account, which is especially advantageous in the presence of noisy data (where an observation may miss its true bin and instead fall in a neighbored bin). In the following, we present some measures of this kind.

- *Quadratic Form Distance:* Given an order $b_1 < b_2 < \ldots < b_n$ on $\mathcal{B}$, a histogram can be written as a vector

$$\vec{h} = (h(b_1), h(b_2), \ldots, h(b_n))^T \ . \tag{2}$$

  Using this representation, the quadratic form distance is defined as

$$d_{QF}(g,h) = \sqrt{(\vec{g}-\vec{h})^T A (\vec{g}-\vec{h})} \ ,$$

  where A is a matrix whose entries $a_{i,j}$ specify the similarity between bins b_i and b_j. Defining the distance $d_{i,j}$ between bin b_i and b_j by the distance between the corresponding cores (mid-points of intervals in the non-fuzzy case), i.e., $d_{i,j} = |i-j|$, we follow [15] and let

$$a_{i,j} = 1 - \frac{d_{i,j}}{\max_{i,j}\{d_{i,j}\}} \ .$$

  As can be seen, (2) performs an all-vs-all comparison of bins, weighting the comparison between b_i and b_j by $a_{i,j}$.

- *Cumulative Distributions:* Another possibility to exploit an order on $\mathcal{B}$ is to replace the original histogram h by the corresponding cumulative distribution, defined by

$$H(b) \ = \sum_{a \le b} h(a) \ ,$$

  and to measure the distance on these cumulative distributions. Specifically, the L_1-norm

$$d_M(g,h) = \sum_{b \in \mathcal{B}} |G(b) - H(b)|$$

$$d_{EMD}(g,h) = \begin{cases} \min\left\{\sum_{\mathcal{B}_n} f_{i,k} \,|\{f_{i,k} : (i,k) \in \mathcal{B}_n\}\right\} & \\ \text{subject to:} & \\ \quad \sum_{k:(i,k)\in\mathcal{B}_n}(f_{i,k} - f_{k,i}) = g(b) - h(b) & \forall\, b \in \mathcal{B} \\ \quad f_{i,k} \geq 0 & \forall\, (i,k) \in \mathcal{B}_n \end{cases} \tag{3}$$

is called the *match distance* [17], and the L_∞-norm

$$d_{KS}(g,h) = \max_{b\in\mathcal{B}}\{|G(b) - H(b)|\}$$

the *Kolmogorov-Smirnov Distance*.

- *Earth Mover's Distance:* The so-called "Earth Mover's Distance" (EMD) is based on the metaphor of moving masses (of earth) from one bin to another one, measuring the corresponding amount of work in terms of the product of mass and distance. The distance between two histograms is then defined by the minimum amount of work needed to transform the first histogram into the second one.

  It is not difficult to see that the problem of computing such a distance can be formalized as a min-flow problem (answering the question which part of the mass $g(b_i)$ should be moved to $h(b_j)$ and vice versa) and, therefore, takes the form of a quadratic program (QP).

  The original problem formulation has a rather high memory requirement, due to the need to store a large number of constraints, which is problematic in our case. Fortunately, [13] proposed an efficient algorithm that makes our problem amenable to the EMD. Using the L_1-norm as ground distance on $\mathcal{B}$, the problem of calculating the EMD still remains a QP, however, with a formulation that is much more compact. The corresponding program, given in eqn. (3), can again be solved with standard QP-solvers. The idea behind the simplification is that it suffices to consider the so-called neighbor-flows between adjacent bins, since all other flows can be replaced by a cost-equivalent sequence of neighbor-flows. Therefore, the QP only considers flows $f_{i,k}$ with $|i - k| = 1$.

5 Experimental Studies

The assessment of a similarity (distance) measure for biomolecular structures, such as protein binding sites, is clearly a non-trivial problem. In particular, since the concept of similarity by itself is rather vague, it is difficult to evaluate corresponding measures in an objective way. To circumvent this problem, we propose to evaluate similarity measures in an indirect way, namely by means of their performance in the context of nearest neighbor (NN) classification. The

underlying idea is that, the better a distance measure is, the better the predictive performance of an NN classifier (using this measure for determining similar cases) should be. More specifically, we shall measure performance by means of a leave-one-out cross validation procedure on a two-class data set, to be introduced next.

5.1 Data

One important problem in pharmaceutical chemistry is the identification of protein binding sites that bind a certain ligand. We selected two classes of binding sites that bind, respectively, to NADH or ATP. This gives rise to a binary classification problem: Given a protein binding site, predict whether it binds NADH or ATP.

More concretely, we compiled a set of 355 protein binding pockets representing two classes of proteins that share, respectively, ATP and NADH as a cofactor. To this end, we used CavBase to retrieve all known ATP and NADH binding pockets that were co-crystallized with the respective ligand. Subsequently, we reduced the set to one cavity per protein, thus representing the enzymes by a single binding pocket. As protein ligands adopt different conformations due to their structural flexibility, it is likely that the ligands in our data set are bound in completely different ways, hence the corresponding binding pocket does not necessarily share much structural similarity. We thus had to ensure the selection of binding pockets with ligands bound in similar conformation. To achieve this, we used the Kabsch algorithm [10] to calculate the root mean squared deviation (RMSD) between pairs of ligand structures. Subsequently, we combined all proteins whose ligands yielded a RMSD value below a threshold of 0.4, thereby ensuring a certain degree of similarity. This value was chosen as a trade-off between data set size and similarity. Eventually, we thus obtained a two-class data set comprising 214 NADH-binding proteins and 141 ATP-binding proteins.

5.2 Methods

As mentioned above, we use a k-nearest neighbor (k-NN) classifier (with different values k) combined with the different types of distance measures introduced in the previous sections. That is, we tried both types of histogram representations introduced in Section III and combined them with the bin-by-bin and cross-bin distance measures discussed in Section IV.

For comparison, we also applied a number of state-of-the art approaches for protein structure comparison to the same problem, including

- kernel methods: the shortest path (SP) kernel [1], the random walk (RW) kernel [7] and the fingerprint (FP) kernel [6];
- graph-based methods: the iterative graph alignment (IGA) [20] and the evolutionary graph alignment (GAVEO) [5];

- geometric approaches: the labeled point cloud superposition (LPCS) [4].

As performance criteria, we were first of all interested in the accuracy of the methods in terms of their classification rates but also measured their efficiency in terms of runtime.

5.3 Results of Comparative Methods

As a point of departure, Table 1 summarizes the results of the approaches used for comparison. As can be seen, there are clear differences in terms of performance: The highest classification accuracy is achieved by LPCS, followed by the fingerprint kernels. The graph-alignment methods (IGA and GAVEO) perform less strongly, and the worst classification rates are produced by the graph kernels.

The runtime reported in the table includes the time needed for an all-against-all comparison of the 355 structures and the time needed to perform a leave-one-out cross validation. As can be seen, all methods require at least one day. Upon closer inspection, however, one can recognize significant differences: On a single-core machine, the fingerprint kernel is the fastest method and needs around 1.5 days, whereas GAVEO needs more than half a year. Needless to say, these methods are not practicable for larger data sets.

Table 1: Classification rates and runtime in hours of a k-NN classifier using different values of k and different distance measures: random walk kernel (RW), shortest path kernel (SP), labeled point cloud superposition (LPCS), fingerprint kernel (FP), iterative graph alignment (IGA), and evolutionary graph alignment (GAVEO).

k	RW	SP	LPCS	FP	IGA	GAVEO
1	0.597	0.606	0.935	0.842	0.766	0.789
3	0.597	0.628	0.916	0.882	0.718	0.766
5	0.597	0.634	0.890	0.873	0.724	0.780
7	0.608	0.625	0.885	0.859	0.718	0.786
9	0.608	0.634	0.862	0.836	0.713	0.766
runtime (h)	1149.88	171.14	361.58	35.98	2136.88	> 5000

5.4 Results for Representation I

Recall that, for the first histogram representation, a protein structure is represented in terms of two histograms, one for the pseudocenters and one for the pairwise distances between these centers. In a first test, we tried these two histograms separately.

Thus, we first reduced the comparison of protein structures to the comparison of their respective pseudocenter histograms, using different bin-by-bin distance measures. The results are shown in Table 2. In light of their simplicity, all variants perform surprisingly well. Moreover, less than 5 seconds are needed

Table 2: Classification rates of bin-by-bin measures on the NADH/ATP data using pseudocenter histograms.

k	d_{MF} $p=1$	d_{MF} $p=2$	d_{MF} $p=\infty$	d_J
1	**0.7831**	0.7239	0.7099	0.7577
3	0.7549	0.7155	0.6986	0.7493
5	0.7268	0.7099	0.6986	0.7211
7	0.7268	0.7070	0.6930	0.7324
9	0.7155	0.7127	0.7099	0.7211
runtime (h)	< 0.001	< 0.001	< 0.001	< 0.001

for the whole all-against-all comparison, so this approach is much faster than the methods used for comparison (due to consistency, the time in again report in hours).

The results obtained by considering only the distance histogram are summarized in Table 3. As can be seen, the performance is significantly worse, suggesting that the geometrical structure of a binding site is less informative than its physico-chemical composition. Moreover, the approach is computationally more expensive, due to the quadratic number of distances and the use of cross-bin measures. Still, however, the runtime remains below 10 minutes, except for the EMD.

Table 3: Classification rates of cross-bin measures on the NADH/ATP data set using distance histograms of type I.

k	d_{QF}	d_M	d_{KS}	d_{EMD}
1	0.597	0.594	0.589	0.665
3	0.555	0.623	0.614	0.676
5	0.569	0.611	0.645	0.685
7	0.580	0.639	0.665	**0.687**
9	0.625	0.654	0.656	0.673
runtime (h)	0.097	0.098	0.084	0.554

Finally, we combined the two best distance measures on pseudocenters and distances, respectively, in terms of the L_2-norm. Thus, we used the Minkowski distance to compare pseudocenter histograms and the earth-mover distance to compare distance histograms. Since the time to calculate the L_2-norm can be neglected, the runtime can still be taken from Table 3. Using this approach, classification rates of more than 80% can be achieved, as can be seen in Table 4.

5.5 Results for Representation II

For the second histogram representation, we used (1) to combine the individual distances on the 28 histograms. As expected, the cross-bin measures, for which

Table 4: Classification rates using representation I and the measures d_{MF} and d_{EMD}, respectively, on the NADH/ATP data set.

k	1	3	5	7	9
accuracy	0.797	**0.837**	0.806	0.806	0.789

Table 5: Classification rates of bib-by-bin measures on the NADH/ATP data set. Combination of all 28 distances by calculating the L_2-norm.

	d_{MF}			d_J
k	$p=1$	$p=2$	$p=\infty$	
1	**0.870**	0.848	0.806	0.873
3	0.854	0.834	0.817	0.870
5	0.794	0.814	0.792	0.848
7	0.769	0.803	0.786	0.842
9	0.763	0.780	0.780	0.834
runtime (h)	0.471	0.473	0.473	0.472

the results are summarized in Table 6, perform somewhat better than the bin-by-bin measures whose results are given in Table 5. As can be seen, the more complex histogram representation leads to a further gain in accuracy, albeit at the expense of a slightly increased runtime. In comparison to existing (graph-based and geometric) methods, however, the runtime is still much smaller, by a factor of about 800. At the same time, this representation achieves even higher classification rates.

5.6 Results for Fuzzy Histograms

Finally, the results on fuzzy histograms are summarized in Table 7 (bin-by-bin measures) and Table 8 (cross-bin-measures). Somewhat surprisingly, cross-bin measures do not produce better results than simple bin-by-bin measures in the fuzzy case. As a possible explanation, note that the boundary problem is already solved by the fuzzy extension itself, making fuzzy histograms much more robust toward noise. Anyway, the combination of fuzzy histograms and bin-by-

Table 6: Classification rates of cross-bin measures on the NADH/ATP data set. Combination of all 28 distances by calculating the L_2-norm.

k	d_{QF}	d_M	d_{KS}	d_{EMD}
1	0.862	0.865	0.859	0.772
3	0.856	**0.882**	0.854	0.749
5	0.845	0.865	0.837	0.732
7	0.823	0.851	0.814	0.738
9	0.823	0.837	0.817	0.721
runtime (h)	0.785	0.470	0.472	11.53

Table 7: Classification rates of bib-by-bin measures on the NADH/ATP data set using fuzzy histograms.

k	d_{MF} $p=1$	d_{MF} $p=2$	d_{MF} $p=\infty$	d_J
1	**0.890**	0.870	0.868	0.885
3	**0.890**	0.873	0.851	**0.890**
5	0.876	0.854	0.814	0.862
7	0.848	0.839	0.811	0.851
9	0.831	0.831	0.797	0.842
runtime (h)	2.21	2.22	2.21	2.22

Table 8: Classification rates of cross-bin measures on the NADH/ATP data set using fuzzy histograms.

k	d_{QF}	d_M	d_{KS}	d_{EMD}
1	0.853	0.862	0.856	0.839
3	0.851	**0.879**	0.865	0.839
5	0.848	0.856	0.851	0.834
7	0.828	0.845	0.825	0.811
9	0.825	0.834	0.828	0.792
runtime (h)	2.52	2.21	2.22	13.15

bin measures seems to be the most promising one, at it achieves the highest classification rates (about 90%). Besides, it remains extremely efficient from a computational point of view; of course, the runtime increases in comparison to the non-fuzzy case since more arithmetical operations must be performed. However, in comparison to existing (graph-based) methods in this field, fuzzy histograms are still more efficient by a factor 85.

6 Conclusions

Returning to the question raised in the introduction to this paper, namely the question concerning the potential usefulness of histogram-based similarity (distance) measures for protein structure comparison, our empirical results clearly provide evidence in favor of an affirmative answer. The best combination of histogram representation and distance measure is able to outperform, in terms of classification accuracy, several state-of-the-art methods in this field and comes close to the best among these methods. The use of fuzzy instead of conventional histograms turned out to be beneficial in this regard. At the same time, our approach is computationally much more efficient. Doubtlessly, it thus provides a viable alternative for large-scale studies in which efficiency is a crucial issue and important prerequisite.

Admittedly, however, our evidence is not yet as thorough as it should be, since the experiments were restricted to a single data set. On the one hand, it is

true that, while sequence data abounds, the collection of data sets in structural bioinformatics is much more difficult, all the more if specific requirements have to met (like those resulting from our experimental design). On the other hand, complementing our experiments by further studies of similar kind is clearly necessary and therefore on the agenda for future work.

References

[1] K. M. Borgwardt and H. P. Kriegel. Shortest-path kernels on graphs. In *International Conference on Data Mining*, pages 74–81, Houston, Texas, 2005.

[2] H. Bunke, X. Jiang, and A. Kandel. On the Minimum Common Supergraph of two Graphs. *Computing*, 65(1):13–25, 2000.

[3] H. Bunke and K. Shearer. A Graph Distance Metric Based on the Maximal Common Subgraph. *Pattern Recognition Letters*, 19(3-4):255–259, 1998.

[4] Thomas Fober and Eyke Hüllermeier. Fuzzy modeling of labeled point cloud superposition for the comparison of protein binding sites. In *IFSA World Congress, EUSFLAT World Conference*, pages 1299–1304, Lisboa, Portugal, 2009.

[5] Thomas Fober, Marco Mernberger, Gerhard Klebe, and Eyke Hüllermeier. Evolutionary construction of multiple graph alignments for the structural analysis of biomolecules. *Bioinformatics*, 25(16):2110–2117, 2009.

[6] Thomas Fober, Marco Mernberger, Vitalik Melnikov, Ralph Moritz, and Eyke Hüllermeier. Extension and empirical comparison of graph-kernels for the analysis of protein active sites. In *Lernen, Wissen, Adaptivitt*, pages 30–36, Darmstadt, Germany, 2009.

[7] Thomas Gärtner. A survey of kernels for structured data. *SIGKKD Explorations*, 5(1):49 – 58, 2003.

[8] Thomas Gärtner. *Kernels for structured data*. World Scientific, Singapore, 2008.

[9] M. Hendlich, F. Rippmann, and G. Barnickel. LIGSITE: Automatic and efficient detection of potential small molecule-binding sites in proteins. *Journal of Molecular Graphics and Modelling*, 15:359–363, 1997.

[10] Wolfgang Kabsch. A solution of the best rotation to relate two sets of vectors. *Acta Crystallographica*, 32:922–923, 1976.

[11] Risi Kondor and Karsten M. Borgwardt. The skew spectrum of graphs. In *International Conference on Machine Learning*, pages 496–503, 2008.

[12] Michael Neuhaus and Horst Bunke. *Briding the Gap between Graph Edit Distance and Kernel Machines*. World Scientific, New Jersey, 2007.

[13] Kazunori Okada and Haibin Ling. An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):840–853, 2007.

[14] Isidore Rigoutsos and Haim Wolfson. Geometric hashing. *IEEE Computational Science Engineering*, 4:1070–9924, 1997.

[15] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[16] S. Schmitt, D. Kuhn, and G. Klebe. A new method to detect related function among proteins independent of sequence and fold homology. *Journal of Molecular Biology*, 323(2):387–406, 2002.

[17] H.C. Shen and A.K.C. Wong. Generalized texture representation and metri. *Computer, Vision, Graphics, and Image Processing*, 23:187–206, 1983.

[18] Sven Siggelkow and Hans Burkhardt. Improvement of histogram-based image retrieval and classification. In *16th International Conference on Pattern Recognition*, volume 3, page 30367, 2002.

[19] Constantin Vertan and Nozha Boujemaa. Using fuzzy histograms and distances for color image retrieval. In *Challenge of Image Retrieval*, pages 1–6, Brighton, United Kingdom, 2000.

[20] N. Weskamp, E. Hüllermeier, D. Kuhn, and G. Klebe. Multiple graph alignment for the structural analysis of protein active sites. *IEEE Transactions on Computational Biology and Bioinformatics*, 4(2):310–320, 2007.