Visibility Graph Network of Multidimensional Time Series Data for Protein Structure Classification

Michaela Areti Zervou[1,2], Effrosyni Doutsi[1], Panagiotis Tsakalides[1,2]
[1]*Institute of Computer Science, Foundation for Research and Technology-Hellas, Heraklion, Greece*
[2]*Department of Computer Science, University of Crete, Heraklion, Greece*
E-mails: zervou@ics.forth.gr, edoutsi@ics.forth.gr, tsakalid@ics.forth.gr

***Abstract*—In the last decades, many studies have explored the potential of utilizing complex network approaches to characterize time series generated from dynamical systems. Along these lines, Visibility Graph (VG) and Horizontal Visibility Graph (HVG) networks have contributed to an important yet difficult problem in bioinformatics, the classification of the secondary structure of low-homology proteins. In particular, each protein is presented as a two-dimensional time series that is later transformed, using either VG or HVG, into two independent graphs. However, this is an inefficient way of processing multidimensional time series as it fails to capture the correlation between the two signals while it also increases the time and memory complexity. To address this issue, this work proposes four novel VG and HVG-based frameworks that are able to deal directly with the multidimensional time series. Each of the methods generates a unique graph following a different visibility rule concerning only the relation between pairs of time series intensities of the multidimensional time series. Experimental evaluation on real protein sequences demonstrates the superiority of our best scheme, with respect to both accuracy and computational time, when compared against the state-of-the-art.**

***Index Terms*—Visibility Graph, Horizontal Visibility Graph, Multidimensional time series, Nonlinear time series analysis, Secondary structure classification**

I. Introduction

Time series analysis comprises statistical models in order to extract meaningful interpretations for analyzing the sample data. Transforming time series into graphs opens up the possibilities of creating fruitful correlations between time series analysis, nonlinear dynamics, and graph theory. The Visibility Graph algorithm (VG) [1] and the Horizontal Visibility Graph algorithm (HVG) [2] are considered two of the most efficient and simple approaches that can be utilized for mapping time series and complex networks [3]–[6]. In particular, they consider the time series points as a sequence of intensity bars that are then connected based on their inter-visibility.

The accelerated evolution in genomics over the past few decades has led to a vast volume of protein sequence evidence from amino acids. The growing need for effective protein secondary structure classification architectures has therefore emerged. Along these lines, numerous studies [6]–[11] demonstrate the potential of transforming the amino acid sequence into a time series and then utilizing complex network methods for the characterization of the arisen dynamical system [6], [11], [12]. Initially, every amino acid in the protein sequence is predicted as one of three secondary structural elements, namely H (helix), E (strand) and C (coil) using the PSI-PRED tool [13]. Then, by employing the Chaos Game Representation (CGR) [14] technique, the updated 3-state sequence, that is of the same length as the original amino acid sequence, is converted into a graphical form where the x- and y-coordinates of each point on the graph are considered as two individual time series that are later analyzed independently with the VG or HVG algorithms.

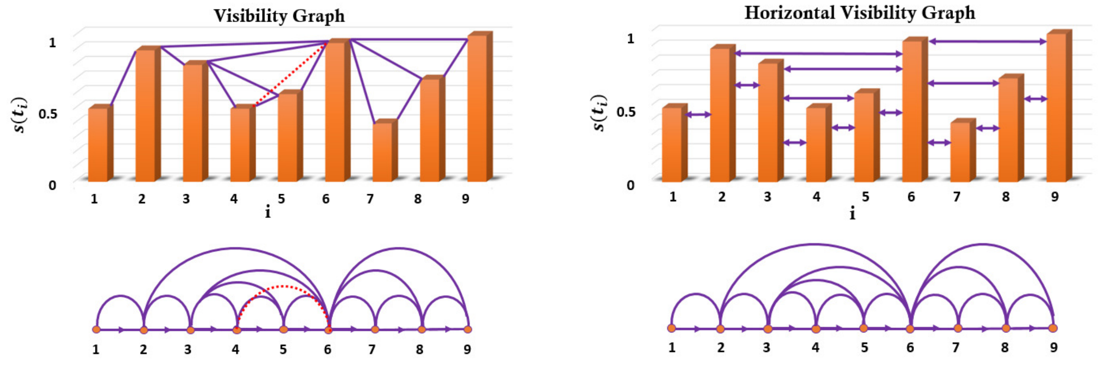
Although the aforementioned architecture leads to high-precision results of the classification of low-homology protein structures [6], [11], [12], their enhanced performance comes at the expense of time and memory complexity due to an inefficient processing of the multidimensional time series data. This is mainly due to the fact that given a two-dimensional time series, (i) the graph analysis algorithm is applied independently to each dimension resulting in a set of two graphs and (ii) a number of f metrics are employed to characterize each graph resulting in a total $2 \times f$ number of features.

To address the above issue, this work processes directly the two-dimensional time series by generating a unique planar graph, considering only the inter-visibility between the pairs of time series intensities among different dimensions, while, other methods generate the entire graph for each of the two dimensions and then apply additional metrics to provide the final graph [15]. Specifically, we propose two independent criteria implying that an edge between two nodes exists if and only if their inter-visibility criterion is satisfied (a) along every time series dimension (the logical AND operation), or (b) in at least one dimension (the logical OR operation). The contributions of this paper are summarized below:

(i) The design of four independent frameworks, namely mdVG-AND, mdVG-OR, mdHVG-AND and mdHVG-OR based on the VG and HVG algorithms, to directly process the two-dimensional time series. Each of the four frameworks is general, scalable, simple to implement, and suitable for the analysis of large, heterogeneous and non-stationary time series.

(ii) The design of a novel feature extraction scheme based on the best performing scenario that enables the discovery of representative patterns, increasing the overall accuracy of the secondary structural class prediction of the protein.

This research work was supported by the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under HFRI faculty grant no. 1725, and by the Stavros Niarchos Foundation within the framework of the project ARCHERS.



(a)

(b)

Fig. 1: Representation of the same time series into a planar graph utilizing (a) Visibility (VG) and (b) Horizontal Visibility Graph (HVG) algorithms. Each t_i value of the time series, represented as an intensity bar, corresponds to a node (orange dot) in the final graph given below. The visibility rule between a pair of nodes is represented as a line between the corresponding time series intensities for the case of VGs and as a horizontal line for the case of HVGs. The dashed red line in (a) highlights the main difference between VG and HVG algorithm when applied on the same time series.

(iii) The significant reduction of the computational complexity of both the feature extraction and the classification processes.

The rest of the paper is organized as follows: Section II is an introduction to the VG and HVG algorithms as well as the metrics utilized to characterize the respective graphs. Section III presents the proposed mdVG and mdHVG architectures that consist of two independent study case scenarios for each algorithm. Section IV describes the evaluation dataset and the classification procedure. Section V represents the experimental results while Section VI draws the conclusion of this work and gives directions for future extensions.

II. Methods

A. Visibility Graphs

A visibility graph (VG) [1] is a graph of inter-visible locations, typically for a set of points and obstacles in the Euclidean plane. Each node in the graph represents a point location, and each edge represents a visible connection between them. Specifically, if the line segment connecting two locations does not pass through any obstacle, an edge is drawn between them in the graph. When the set of locations lies in a line, this can be understood as an ordered series. Visibility graphs have therefore been extended to the realm of time series analysis. To construct a VG, we consider $\{t_i\}_{i=1}^{N}$ as an N time point series in temporal ordering. The VG is obtained by first representing the time series points as $\{n_i\}_{i=1}^{N}$ nodes in a network, where the index of nodes i and j represent times t_i and t_j with intensities $s(t_i)$ and $s(t_j)$, respectively. Two arbitrary data in the time series have visibility and consequently become two nodes that share an edge in the associated graph, if any other intermediate time point t_k, such as that $t_j < t_k < t_i$, has intensity $s(t_k)$ that fulfills,

$$s(t_k) < s(t_i) + (s(t_j) - s(t_i))\frac{t_k - t_i}{t_j - t_i} \qquad (1)$$

A VG representation is given in Fig. 1(a). Some basic properties of the mapping include undirectedness, connectedness and invariance under affine transformations. Moreover, in this work the resulted graph is unweighted. Finally, the properties of the resulted graph $G = (V, E)$, $V = |N|$, $E = |M|$, with N and M being the number of the nodes and edges respectively, are represented by the following nine measures (ref. [11] for the mathematical definitions) that are later employed for classification purposes,

- **Maximum Degree:** The degree of a node is the number of edges connected to the node.
- **Average shortest path:** The average path length between two vertices is the shortest distance among them.
- **Diameter:** The diameter is a measure of the compactness in a network. Practically, is the longest shortest path between any two nodes in the network
- **Clustering coefficient:** The clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together.
- **Energy:** The energy of a network is defined as the sum of the adjacency matrix's eigenvalues.
- **Laplacian Energy:** The Laplacian energy is a proper extension of the graph-energy concept.
- **Pearson correlation coefficient:** To understand whether an unweighted undirected network is of assortive or disassortive type, the Pearson correlation coefficient of the degrees at either ends of an edge is calculated.
- **Average closeness centrality:** The closeness value is the inverse of the average distance between two nodes.
- **Number of nodes:** The number of nodes is an important feature for the network and it is equal to N.

B. Horizontal Visibility Graphs

A simple and fast computational method, known as horizontal visibility graph (HVG) [2], maps time series into graphs. HVG is invariant under affine transformations of the series

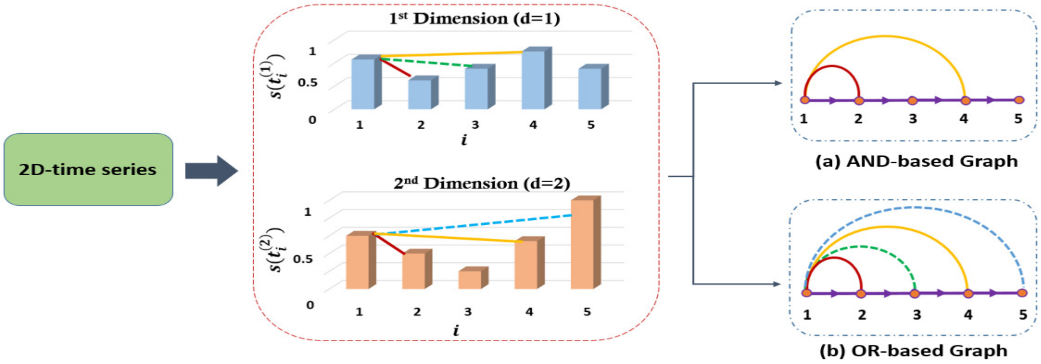


Fig. 2: A general example of the proposed (a) AND-based and (b) OR-based scenarios described in Sec. III with the resulting graph for both VG and HVG algorithms being equal. Suppose a two-dimensional time series. For each dimension $d = \{1, 2\}$ the time instance $t_i^{(d)}$, with intensity $s(t_i^{(d)})$, represents the index of node i. This example focuses only on the visibility between the first time instance with the rest. Visibility that is met in both dimensions between the same pair of time instances is drawn with the same colored edge, otherwise the edge color and shape is different. When two time instances share an edge in both dimensions an AND-based graph arises (a), whereas an OR-based graph results when edges between two intensities meet in at least one dimension (b).

data and its main focus lies on the time series structural properties (periodicity, fractality, etc.). Specifically, let a time series $\{t_i\}_{i=1}^N$, the HVG algorithm assigns each sample point t_i, with intensity $s(t_i)$, as a node in a graph G. Then, two nodes in the network are connected if any other intermediate time point t_k, such as that $t_j < t_k < t_i$, has intensity $s(t_k)$ that satisfies the following geometrical rule,

$$s(t_k) < \min\{s(t_i), s(t_j)\}, \quad i < k < j \qquad (2)$$

In essence, two nodes n_i and n_j share an edge when a horizontal line can be drawn among them without intersecting, in terms of magnitude, any intermediate node, as presented in Fig. 1(b). In this study, the resulted horizontal visibility graph is undirected, unweighted and connected. Finally, the features described in Section II-A can be utilized to form the feature matrix that will be later serve the classification procedure.

III. Proposed Architecture

In the herein work we propose four network analysis architectures based on the VG and HVG algorithms, namely, mdVG-AND, mdVG-OR, mdHVG-AND and mdHVG-OR, that are directly applicable to multidimensional time series data. It is important to note that the novel feature of the herein work is that instead of constructing the entire graph for each of the two dimensions and then applying various techniques to yield a final graph as in [15], we consider only the visibility between pairs of time series intensities per dimension. In particular, consider a multidimensional time series $\{t_i^{(d)}\}_{i=1}^N$ with d=1,..,D being the number of dimensions. A planar mdHVG-AND graph with a set of edges E, is constructed based on the following rule

- Two nodes n_i and n_j share an edge $e_{ij} \in E$ in the mdHVG-AND graph when for each intermediate time instance, t_k, $t_i < t_k < t_j$,

  $$E = \{e_{ij} |\ s(t_k^{(d)}) < \min\{s(t_i^{(d)}), s(t_j^{(d)})\}, \forall d, d = 1, .., D\}$$

  The mdVG-AND graph shares the same principle where in this case the rule is based on Eq.1.

Similarly, a mdHVG-OR graph with a set of edges E, can be derived when the following rule is satisfied

- Two nodes n_i and n_j share an edge $e_{ij} \in E$ in the mdHVG-OR graph when for each intermediate time instance, t_k, $t_i < t_k < t_j$,

  $$E = \{e_{ij} |\ \exists d, s(t_k^{(d)}) < \min\{s(t_i^{(d)}), s(t_j^{(d)})\}, d = 1, .., D\}$$

  The mdVG-OR graph shares the same principle where the rule is based on Eq.1.

An example of the proposed frameworks is presented in Fig. 2.

The VG and HVG algorithm implementation in this work is based on the fast code versions of Iacobello *et al.* [16]. Comparing the AND and OR-based architectures in terms of computational efficiency, the AND-based approach outperforms the OR-based since once the visibility rule is not met in at least one dimension d for a pair of nodes, the algorithm immediately considers that no edge is shared between them and proceeds to the next pair of nodes, whereas in the case of the AND-based scheme the procedure continues for all the remaining dimensions.

IV. Performance Evaluation

This section describes in detail the dataset employed, the data pre-processing as well as the classification procedure. Every proposed framework is implemented in MATLAB, on a desktop computer equipped with a CPU processor (Intel Core i7-7700) clocked at 2.8GHz, and a 8 GB RAM.

A. Dataset description

This work employs the 25PDB dataset [17] that includes 1673 proteins of varying length with 25% sequence homology. The proteins are categorised based on their structural class with 443 of them belonging to α-fold, 443 to β-fold, 346

FDA												
	VG		**mdVG-AND**		**mdVG-OR**		**HVG**		**mdHVG-AND**		**mdHVG-OR**	
	Sens.	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**
α	80.22%	94.59%	79.14%	98.19%	76.81%	96.67%	82.21%	97.13%	83.59%	97.25%	78.16%	96.47%
β	79.86%	93.02%	69.57%	98.60%	74.01%	93.01%	72.57%	90.66%	69.63%	97.60%	76.37%	96.21%
α/β	66.55%	96.08%	61.40%	94.76%	64.14%	94.36%	67.22%	94.59%	69.58%	95.40%	63.98%	92.67%
$\alpha+\beta$	67.96%	81.37%	84.56%	74.04%	70.35%	78.01%	64.51%	79.86%	80.82%	77.86%	71.58%	78.57%
OA	87.03%		87.18%		85.87%		85.95%		**88.13%**		86.51%	

SVM												
	VG		**mdVG-AND**		**mdVG-OR**		**HVG**		**mdHVG-AND**		**mdHVG-OR**	
	Sens.	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**	**Sens.**	**Spec.**
α	83.79%	94.28%	87.01%	96.52%	80.95%	96.10%	81.21%	96.08%	85.99%	95.85%	89.43%	93.98%
β	80.21%	92.72%	75.25%	97.44%	74.83%	93.86%	73.84%	89.33%	75.03%	96.03%	77.65%	95.91%
α/β	67%	94.65%	66.10%	94.62%	62.27%	95.02%	66%	92.49%	65.72%	94.59%	63.60%	94.55%
$\alpha+\beta$	64.67%	83.89%	77.91%	80.82%	72.65%	79.11%	57.35%	81.67%	73.80%	80.90%	68.97%	82.79%
OA	87.16%		**88.59%**		86.64%		84.91%		87.84%		87.79%	

TABLE I: Performance evaluation of VG, mdVG-AND, mdVG-OR, HVG, mdHVG-AND and mdHVG-OR with SVM and FDA classifiers. The highest achieved accuracy for SVM and FDA per experiment is highlighted and indicated in bold.

to α/β-fold and 441 to $\alpha+\beta$-fold. Instead of dealing with the protein primary structure the PSI-PRED tool [13] is utilized to predict the role of each amino acid in the protein secondary structure. Particularly, PSI-PRED transforms the initial amino acid sequence to a sequence of equal length that now consists of only three states that describe its secondary structure, namely coils (C), strands (E) and helices (H). This simplification not only reduces the dimensionality of our data from 20 amino acids to three structural elements but also the overall computational complexity. Thereafter, in order to transform a unidimensional sequence of secondary structural elements into a two-dimensional time series, CGR is employed that yield a two-dimensional time series [8]–[12].

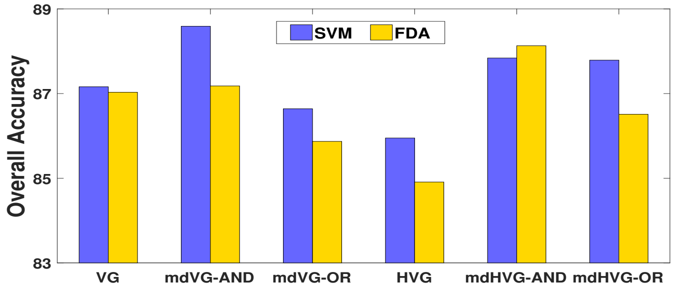
B. *Classification*

In previous studies [6]–[9], [12], the leave-one-out cross-validation process is employed for the specific classification problem. However, based on our experiments, a 10-fold-cross-validation yields approximately the same overall accuracy in considerably less time. Therefore, in this work the data are randomly split into 10-folds for training and testing and the procedure is repeated 100 times. In more detail, the feature matrix is initially z-score normalized and then, the performance of each architecture is evaluated by two well known classifiers independently. Particularly, a Gaussian-kernel Support Vector Machine (SVM) as well as Fisher's Linear Discriminant Analysis (FDA) are applied separately on the normalized feature matrix for discriminating between the four secondary structural classes α-fold, β-fold, α/β-fold and $\alpha+\beta$-fold. For the Gaussian SVM classifier, the regularization parameter C and kernel width parameter γ can take all positive values log-scaled in the range $[10^{-3}, 10^{3}]$.

V. Experimental Results

The performance of the proposed VG, HVG, mdVG-AND, mdVG-OR, mdHVG-AND and mdHVG-OR frameworks is evaluated in terms of sensitivity, specificity and Overall Accuracy (OA) for SVM and FDA, respectively. As depicted in Table I, proteins that belong to the α-fold and β-fold class are better predicted in most case studies for both SVM and FDA classifiers, whereas the α/β-fold proteins meets the lowest prediction accuracy since it is difficult to differentiate them from the others. Furthermore, it is demonstrated that the original VG approach outperforms HVG for both SVM and FDA classifiers. Following that, based on the bar figure in Table II we observe that in the instance of mdVG-AND with SVM classifier the overall accuracy increases significantly, as opposed to the mdVG-OR scheme, where the performance drops for both classifiers. In terms of the mdHVG-AND framework, it is given that the FDA classifier achieves better results compared to both SVM classifier as well as the remaining approaches. On the other side, the mdHVG-OR scheme outperforms the HVG algorithm, yet not as well as the mdHVG-AND architecture. Overall, the most efficient framework in terms of classification accuracy is the mdVG-AND with SVM classifier. In this vein, it is worth noting that, in general, AND-based schemes outperform all the other schemes (original VG, original HVG and OR-based) in terms of classification performance. This tendency may be explained by the fact that, unlike the other techniques, AND-based architectures exploit both intra- and inter-data correlations between all the different dimensions of the time series.

The summarized predicted results for the 25PDB dataset are given in Table II. It is important to note that the provided Computational Complexity (CC) is decoupled from the classification procedure. As previously stated, by constructing the VG or HVG network, a protein is represented as a real-valued vector with 17 characteristics, whereas for the cases of mdVG-AND, mdVG-OR, mdHVG-AND and mdHVG-OR the protein is characterized by a real-valued vector of 9 features in total. As indicated, the herein proposed multidimensional architectures achieve similar or even higher classification accuracy while extracting the minimum number of features which in turn not only reduces the memory complexity of the system but the computational complexity as well. Moreover,



	Number of Features	Overall Accuracy		CC (minutes)
		SVM	FDA	
VG	17	**87.16%**	87.03%	1.45
mdVG-AND	9	**88.59%**	87.18%	0.77
mdVG-OR	9	**86.64%**	85.87%	1.08
HVG	17	**85.95%**	84.91%	1.17
mdHVG-AND	9	87.84%	**88.13%**	**0.70**
mdHVG-OR	9	**87.79%**	86.51%	1.46

TABLE II: Summarized predicted results for the 25PDB dataset. The overall accuracy for SVM and FDA classifiers is presented in the bar figure. The highest achieved accuracy for SVM and FDA per experiment is indicated in bold on the respective table, and the most efficient architecture in terms of classification accuracy and computational complexity (CC) considering only the feature extraction procedure is highlighted in blue.

it is clear that SVM classifier outperforms FDA classifier in most cases with the highest overall accuracy (88.59%) being achieved with the mdVG-AND scheme that requires only 0.77 minutes to extract 1673×9 features. Nevertheless, SVM is a computationally expensive classifier that, as opposed to FDA, it demands hyperparemeter tuning. In particular, SVM requires approximately 3.24 min to perform a 10-fold-cross validation of 1673 samples, whereas FDA is executed for the same setup in approximately 0.17 minutes. Therefore, FDA is considered as a more efficient classifier in terms of computational complexity. In this regard, the mdHVG-AND framework with FDA achieves an insignificantly lower overall accuracy (88.13%) when compared to the best performing scheme, namely the mdVG-AND scheme with SVM (88.59%), but yet in importantly lower total computational cost when both the feature extraction and classification procedures are taken into account (4.01 minutes vs 0.87 minutes, respectively). Therefore, the mdHVG-AND architecture with FDA classifier is considered as the optimal to be employed for classifying protein secondary structures.

VI. Conclusions and future work

This work designs and implements four efficient and novel approaches, namely mdVG-AND,mdVG-OR, mdHVG-AND and mdHVG-OR, for transforming a multidimensional time series into a unique planar graph based on the VG and HVG algorithms. Each of the four frameworks is simple to implement, general, scalable, and suitable for the analysis of large, heterogeneous and non-stationary time series. The study on real protein data revealed the superiority of the mdHVG-AND scheme combined with an FDA classifier, as compared to the state-of-the-art in terms of overall classification accuracy, feature multitude and running time complexity. The proposed scheme can be viewed as an efficient protein secondary structure classification architecture.

An extension of this work will consider a deep learning framework for protein structure classification based on the graph representation of the proteins resulted with the utilization of the herein proposed mdHVG-AND algorithm.

References

[1] L. Lacasa, B. Luque, F. Ballesteros, J. Luque, and J. C. Nuno, "From time series to complex networks: The visibility graph," *Proc. of the National Academy of Sciences*, vol. 105, no. 13, pp. 4972–4975, 2008.

[2] B. Luque, L. Lacasa, F. Ballesteros, and J. Luque, "Horizontal visibility graphs: Exact results for random time series," *Physical Review E*, vol. 80, no. 4, p. 046103, 2009.

[3] Z.-K. Gao, Q. Cai, Y.-X. Yang, N. Dong, and S.-S. Zhang, "Visibility graph from adaptive optimal kernel time-frequency representation for classification of epileptiform eeg," *International Journal of Neural Systems*, vol. 27, no. 04, p. 1750005, 2017.

[4] R. Flanagan and L. Lacasa, "Irreversibility of financial time series: a graph-theoretical approach," *Physics Letters A*, vol. 380, no. 20, pp. 1689–1697, 2016.

[5] S. Sannino and et al., "Visibility graphs for fmri data: Multiplex temporal graphs and their modulations across resting-state networks," *Network Neuroscience*, vol. 1, no. 3, pp. 208–221, 2017.

[6] Z. Zhao, L. Luo, and X. Liu, "Low-homology protein structural class prediction from secondary structure based on visibility and horizontal visibility network," *American Journal of Biochemistry and Biotechnology*, vol. 14, pp. 67–75, 2018.

[7] J.-Y. Yang and et al., "Prediction of protein structural classes by recurrence quantification analysis based on chaos game representation," *Journal of Theoretical Biology*, vol. 257, no. 4, pp. 618–626, 2009.

[8] J.-Y. Yang, Z.-L. Peng, and X. Chen, "Prediction of protein structural classes for low-homology sequences based on predicted secondary structure," *BMC bioinformatics*, vol. 11, no. 1, p. S9, 2010.

[9] M. H. Olyaee, A. Yaghoubi, and M. Yaghoobi, "Predicting protein structural classes based on complex networks and recurrence analysis," *Journal of theoretical biology*, vol. 404, pp. 375–382, 2016.

[10] M. Zervou, E. Doutsi, P. Pavlidis, and P. Tsakalides, "Efficient dynamic analysis of low-similarity proteins for structural class prediction," in *28th European Signal Proc. Conf. (EUSIPCO)*. IEEE, 2020, pp. 1328–1332.

[11] M. A. Zervou, E. Doutsi, P. Pavlidis, and P. Tsakalides, "Structural classification of proteins based on the computationally efficient recurrence quantification analysis and horizontal visibility graphs," *bioRxiv*, pp. 2020–10, 2021.

[12] H. Jiang, A. Zhang, Z. Zhang, Q. Meng, and Y. Li, "Protein tertiary structure prediction based on multiscale recurrence quantification analysis and horizontal visibility graph," in *International Symposium on Neural Networks*. Springer, 2019, pp. 531–539.

[13] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *Journal of molecular biology*, vol. 292, no. 2, pp. 195–202, 1999.

[14] H. J. Jeffrey, "Chaos game representation of gene structure," *Nucleic acids research*, vol. 18, no. 8, pp. 2163–2170, 1990.

[15] J. Liu and Q. Li, "Planar visibility graph network algorithm for two dimensional timeseries," in *2017 29th Chinese Control And Decision Conference (CCDC)*. IEEE, 2017, pp. 1352–1357.

[16] G. Iacobello and et al., "Visibility graph analysis of wall turbulence time-series," *Physics Letters A*, vol. 382, no. 1, pp. 1–11, 2018.

[17] http://biomine.cs.vcu.edu/datasets/SCPRED/SCPRED.html.