Visibility graph for time series prediction and image classification: a review

Tao Wen · Huiling Chen · Kang Hao Cheong

Received: 29 May 2022 / Accepted: 9 October 2022 / Published online: 31 October 2022 © The Author(s), under exclusive licence to Springer Nature B.V. 2022

Abstract The analysis of time series and images is significant across different fields due to their widespread applications. In the past few decades, many approaches have been developed, including data-driven artificial intelligence methods, mechanism-driven physical methods, and hybrid mechanism and data-driven models. Complex networks have been used to model numerous complex systems due to its characteristics, including time series prediction and image classification. In order to map time series and images into complex networks, many visibility graph algorithms have been developed, such as horizontal visibility graph, limited penetrable visibility graph, multiplex visibility graph, and image visibility graph. The family of visibility graph algorithms will construct different types of complex networks, including (un-) weighted, (un-) directed, and (single-) multi-layered networks, thereby focusing on different kinds of properties. Different types of visibility graph algorithms will be reviewed in this paper. Through exploring the topological structure and information in the network based on statistical physics, the property of time series and images can be discovered. In order to forecast (multivariate) time series, several variations of local random walk algorithms and different information fusion approaches are applied to measure the similarity between nodes in the network. Different forecasting frameworks are also proposed to consider the information in the time series based on the similarity. In order to classify the image, several machine learning models (such as support vector machine and linear discriminant) are used to classify images based on global features, local features, and multiplex features. Through various simulations on a variety of datasets, researchers have found that the visibility graph algorithm outperformed existing algorithms, both in time series prediction and image classification. Clearly, complex networks are closely connected with time series and images by visibility graph algorithms, rendering complex networks to be an important tool for understanding the characteristics of time series and images. Finally, we conclude in the last section with future outlooks for the visibility graph.

Keywords Time series prediction · Image classification · Visibility graph · Complex network

1 Introduction

With the growth of the Internet and mobile technology, the amount of data faced by researchers is ever-increasing [1, 2]. Describing big data and the complicated relationships between data has become an open problem that still needs to be solved. Artificial intel-
Ligence and machine learning algorithms [3,4], due to their data-driven characteristics, have become one of the most popular algorithms in various fields because they can discover the complicated information behind big data and serve as a black-box model to address problems that could not be solved in the past. In the field of data mining, time series and image analysis are indispensable topics, attracting researchers in various fields [5,6], including computer science [7], economics [8], environment [9], and biology [10]. Researchers have proposed various algorithms to promote the development and progress of this field [11–13], including data-driven models, mechanism-driven models, and hybrid mechanisms and data-driven models. For example, a long short-term memory network with a chaotic Henry gas solubility optimization model was developed to forecast the crude oil time series [14], a nonlinear hybrid model was applied to analyze and predict the wind speed data collected from five wind farms in the Marmara region, Turkey [15], a nonlinear autoregressive network with exogenous inputs is applied to analyze the impact of environmental stresses on the infection of COVID-19 [16], recurrent neural networks based on the dual-stage two-phase model were proposed to address long-term multivariate time series prediction [17], different divergence and entropy are applied to analyze the multiscale structure and irreversibility in financial time series [18, 19], and a hybrid model consisting of 2D curvelet transform and deep learning technique was developed to diagnose COVID-19 disease based on chest X-ray images [20].

A complex system includes many components that are related to each other. Complex systems have been used to address practical problems [21,22], including social media, modern cities, human brains, infrastructure systems, and economic organizations. Complex systems are often modeled by complex networks. Nodes and edges in the network usually represent individuals (components) and connections between them in the system [23,24], respectively. Complex networks have been widely applied, for instance, identifying influential individuals in the spread [25,26], analyzing the dynamics of the ecosystem [27–30], studying synchronization control [31,32], and discovering properties of complex systems [33,34]. Interdisciplinary research has also been carried out to strengthen the ability of complex networks to solve problems in combination with other disciplines, including entropy [35,36], fuzzy theory [37,38], game theory [39,40], machine learning [41], evidence theory [42–44], and information fusion [45–47]. Complexity measure and uncertainty measure are very important for time series, which can be measured by entropy [48,49]. Belief entropy has the efficiency to model uncertainty of mass function [50,51], which is used in time series analysis [52]. Several algorithms have contributed to describing the complicated information and features hidden in time series and images. However, when the complexity of the system increases, the state of the art will not be able to describe the property of time series and images properly due to their own limitations. Complex networks, as one of the most effective interdisciplinary methods for describing complex systems, have been used to analyze time series and images [53,54]. Various algorithms have been developed to construct complex networks from time series [55,56], including proximity networks, visibility graphs, and transition networks. The visibility graph (VG) algorithm [57–59] is an important algorithm that constructs complex networks with corresponding topological structures based on the dynamic property of the univariate/multivariate time series, which has been widely used.

In the network generated by the visibility graph algorithm, whether nodes are connected is determined by the visibility criterion between data points in the time series. In detail, the time series will be displayed through the histogram. If the straight “visibility” line that connects two data points does not intersect any intermediate data height [57], two data points in the time series can see each other from the top of the bar, resulting in the connectivity between two nodes in the natural visibility graph. Luque et al. [60] then developed the horizontal visibility graph. The novelty of this algorithm is that two nodes in the network will be connected if a horizontal line that connects corresponding data points in the time series does not intersect any intermediate data height. Other types of visibility graphs [61] have been proposed to obtain the network with different structure based on the feature of time series, such as (un-) weighted, (un-) directed, and (single- multi-layered networks. Numerous properties in the time series can be revealed by the family of visibility graphs, for example, estimating the Hurst exponent of fractal stochastic processes [62,63], proving the relationship between the power-law degree distribution and fractality in series [57], analyzing multifractal properties of time series [64], and measuring the irreversibility of real-valued time series [65–67].
In addition, it has been applied in different fields to address practical problems, including studying the dynamics of a passive scalar plume [68], planning long-voyage routes [69], analyzing electroencephalogram signals [70], extracting hidden information in coupled timeseries series [71,72], and aggregating data in complex systems [73].

Recently, the family of visibility graph algorithms has been applied to forecast the time series by analyzing the structural information of the network [74]. After the time series \((t_i, y_i), t_i = 1, 2, \ldots, T\) is converted into a complex network, the similarity between nodes is measured by the local random walk algorithm, and the most similar node to the last node \((t_T, y_T)\) in time series is used to forecast the data point at \(T+1\). Apart from the local random walk, there are many algorithms for identifying the similarity between nodes in the network [75], for example, the similarity can be measured by the relative entropy based on the local network [76], and the local random walk approach was improved to be applied in the visibility graph [77]. Fuzzy logic has been applied to determine weight parameters of different sources of information, thereby considering more information in the network to improve the prediction accuracy [78]. The information considered in prediction models includes the information in the last data point \((t_T, y_T)\) [74], the relationship between \((t_T, y_T)\) and its most similar node(s) [79], and the adjacent information [80]. The weight parameters of the prediction model can also be determined by the distance between data points [79,81]. The induced-ordered weighted averaging aggregation (IOWA) operation and network-based multiple time-frequency spaces were applied to forecast time series to improve the accuracy [82]. In order to validate the performance of these visibility graph-based prediction models, both in-sample forecasting and out-of-sample frameworks have been applied. After studying the time series, the image visibility graph [83] has been proposed to map images into complex networks to analyze properties of images, including filters, pattern recognition, and texture classification. The global features, local features, and multiplex features can be extracted from images via the image visibility graph [83]. Notably, the visibility patch can accurately describe the local information of images, thereby embedding the image in a low-dimensional space. The image visibility graph can be further used to classify images into different categories by combining with several machine learning models. Compared with traditional artificial intelligence models, including VGG16, VGG19, and convolutional neural network (CNN), the best classification accuracy obtained by image visibility graph algorithms [83–85] is higher, illustrating the effectiveness of the visibility graph algorithm.

Here, we have reviewed the methods to map time series and images into complex networks, mainly the visibility graph algorithm and its various variants. In addition, we have also reviewed the time series forecasting and image classification methods based on the visibility graph algorithm. These constitute the framework of this review paper. Different kinds of complex network construction methods are first introduced in Sect. 2. The visibility graph and its improved forms are then introduced in Sect. 3, for example, the horizontal visibility graph and multiplex visibility graph are applied to analyze univariate time series and multivariate time series (Sect. 3.1 to Sect. 3.4), respectively. A variety of other improved visibility graph algorithms with different visibility criteria and time complexity is introduced in Sect. 3.5. Analyzing different types and characteristics of time series by the visibility graph are discussed in Sect. 3.6. After mapping the time series into the complex network, the time series can be predicted through the local random walk model and other information fusion methods (Sect. 4), including fuzzy logic, distance-based method, and IOWA operation. Construction Cost Index and several criteria are used to compare the accuracy obtained by VG-based algorithms with other approaches (simple moving average approach). The image visibility graph and its applications in feature extraction, filters, pattern recognition, and texture classification are introduced in Sect. 5. The conclusions and future outlooks are discussed in Sect. 6.

2 Network-based prediction approach

Given a time series \((t_i, y_i), t_i = 1, 2, \ldots, T\) and \(T\) is the length, Zhang et al. [86] developed an algorithm to map the pseudoperiodic time series into a complex network. \((t_i, y_i)\) is first divided into \(|N|\) disjoint cycles \([C_1, C_2, \ldots, C_{|N|}]\) based on the local minimum or local maximum value in the time series. Each node in the network will represent a cycle \(C_i\). There are two ways to determine whether two nodes are connected in this approach: (1) required distance \(\theta\) and (2) lin-
ear correlation coefficient $\rho$. The distance between two nodes ($C_i$ and $C_j$) is defined as follows,

$$D_{ij} = \min_{l=0, 1, \ldots, l_i, l_j} \frac{1}{l_i} \sum_{k=1}^{l_i} \|X_k - Y_{k+l_j}\|,$$  \hspace{1cm} (1)

where $l_i$ and $l_j$ are the length of $C_i$ and $C_j$, respectively. It is important to note that $l_j$ should be larger than $l_i$ in this model. $X_k$ and $Y_k$ are the $k$th point of $C_i$ and $C_j$, respectively. If the distance $D_{ij}$ between two nodes is smaller than $\theta$ (for the toy model) or $\rho$ (for the experimental time series), $C_i$ and $C_j$ will be connected in the complex network. The steps of this mapping model is given in Algorithm 1.

**Algorithm 1** Construct the complex network from the time series [86].

**Input:** Time series $(t_i, y_i)$; Required distance $\theta$ or linear correlation coefficient $\rho$;

**Output:** Complex network $G(V, E)$;

1. Find the local minimum or local maximum value in the time series, and divide the time series into $|N|$ disjoint cycles $\{C_1, C_2, \ldots, C_{|N|}\}$;
2. Each cycle is represented by a node, namely, the network has $|N|$ nodes;
3. for $i = 1$ to $|N|$ do
   4. for $j = 1$ to $|N|$ do
      5. $D_{ij} = \min_{l=0, 1, \ldots, l_i, l_j} \frac{1}{l_i} \sum_{k=1}^{l_i} \|X_k - Y_{k+l_j}\|$;
      6. if $D_{ij} < \theta$ or $D_{ij} < \rho$ then
         7. Node $i$ is connected to node $j$;
      8. end if
   9. end for
10. end for

Xu et al. [87] then developed a new model to construct the network from the time series that can describe several characteristics of continues dynamics (such as periodic, chaotic, and periodic with noise) based on the distribution of sub-networks, and more details about this work can be found in Ref. [87]. The steps are shown below:

Step (1) Obtain the time series $(t_i, y_i)$, where $t_i = 1, 2, \ldots, T$.

Step (2) The series is embedded in a phase space [88], and each state in the phase space is represented as a node in the network.

Step (3) Connect each node to its $\tau$ nearest neighbor nodes to generate the complex network.

For this method, the temporal separation of these eligible neighbor nodes should be larger than the average period of the series to avoid inhabiting the same “strand” [89]. There will be $\tau$ new nearest neighbor nodes for each node, regardless of whether the node has been connected. In addition, there are no multiple links between each pair of nodes. Therefore, each node has an average of $2\tau$ edges connected to neighbor nodes ($|k| = 2\tau$). Selecting $\tau$ nearest neighbor nodes in this method is more flexible than setting the distance threshold, which can obtain an adjustable threshold based on the density of phase space. They also suggest that this approach works best when $\tau$ is $[3, 8]$ and $[3, 4]$ for maps and flows, respectively [87].

Lacasa et al. [57] proposed the visibility algorithm to construct the complex network from the time series based on the visibility criterion. An example time series and corresponding visibility graph (VG) are given in Fig.1a. In this algorithm, each point will be represented by a node in the same order in the network. For any two data points $(t_i, y_i)$ and $(t_j, y_j)$, node $i$ and node $j$ will be connected if visibility exits between data points in time series, that is, any point $(t_k, y_k)$ between them meets the condition,

$$y_k < y_j + (y_i - y_j) \frac{t_j - t_k}{t_j - t_i}. \hspace{1cm} (2)$$

Therefore, the visibility graph has the following three properties,

(1) **Undirected** The edges between nodes have no direction, which is different from the sequential time series.

(2) **Connected** There is no isolated node in the network as the visibility exists between every pair of neighbor data points.

(3) **Invariant under affine transformations** The rescaling and translation of horizontal and vertical axes do not affect the visibility criterion and the structure of the network.

The data in the time series is sequential, and the property of the time series is mainly analyzed from the historical data. The visibility between data points in the time series can be expressed as the connection between nodes, and the properties in the series are also inherited by the network. In detail, periodic series, random series, and fractal series are mapped to regular network, random network, and scale-free network, respectively. The periodic series and regular network are shown in Fig.1a. The degree distributions that can indicate the property of random and scale-free networks constructed by corresponding time series can be found in
Fig. 2. As discussed in ref [57], a natural bridge can be built between the time series analysis and network theory through the visibility graph.

3 Visibility graph

Due to the importance of the visibility graph, the family of the visibility graph is introduced in this section. Different types of networks can be constructed based on different visibility criteria, including (un-) weighted, (un-) directed, and (single-) multi-layered network. Therefore, this section is the foundation for time series prediction and image classification.

3.1 Horizontal visibility graph

Different from the visibility graph, node $i$ and node $j$ will be connected in the horizontal visibility graph (HVG) if all nodes between them ($t_i < t_k < t_j$) meet the following geometrical criterion [60],

$$y_k < y_i, y_k < y_j.$$  \hspace{1cm} (3)

The criterion can also be rewritten as

$$y_k < \inf \{ y_i, y_j \}.$$  \hspace{1cm} (4)

Due to the strict mapping rule of HVG, HVG is a sub-network of VG when they are constructed by the same time series. In addition, the average degree of nodes in HVG is smaller than the average degree of nodes in VG, so the relationship between nodes is limited to the local structural property. An example of HVG is shown in Fig. 1b. In addition to the last two properties of VG, the horizontal visibility graph has three additional properties,

1. Reversible and irreversible properties of the converting
   Information loss in the series caused by the binary adjacency matrix in unweighted networks can be avoided by using weighted networks to represent the time series.

2. Directed and undirected properties of the converting
   Undirected networks are constructed by this algorithm, but directed networks can also be constructed by distinguishing the ingoing degree $d_{in}$ and outgoing degree $d_{out}$.

3. Geometric criteria difference
   HVG has “less visibility” than VG due to its strict geometric criteria, but it does not affect qualitative features of the network.

Based on the characteristic of the HVG, correlated stochastic, uncorrelated, and chaotic processes can be easily classified by the HVG algorithm [91]. Apart from the undirected networks, the directed links (from time $t_i$ to time $t_j > t_i$) are introduced in the network by the directed VG and HVG algorithms [65], which can show the time-reversal asymmetry of the series. The degree $d(t_i)$ of node at time $t_i$ is defined as the sum of the ingoing degree $d_{in}(t_i)$ and outgoing degree $d_{out}(t_i)$, that is, $d(t_i) = d_{in}(t_i) + d_{out}(t_i)$. The irreversibility of the time series can be measured by the Kullback–Leibler divergence between in-degree and out-degree distribution in the directed HVG [65].

3.2 Limited penetrable visibility graph

Based on the framework of VG, the limited penetrable visibility graph (LPVG) [90] was proposed to consider more links when mapping the time series into a complex network. The limited penetrable distance $\lambda$ is defined in this method. Two nodes (node $i$ and node $j$) will be connected if there are only $n < \lambda$ nodes (node $k$) that do not fulfill the condition in Eq. (2), which means there are less than $\lambda$ intermediate data points $(t_k, y_k)$ are

$$y_k > y_j + (y_i - y_j) \frac{t_j - t_k}{t_j - t_i}.$$  \hspace{1cm} (5)

The LPVG has the same three properties as the VG, and each node has at least $2\lambda + 1$ edges. In addition, LPVG has stronger connectivity compared to VG and HVG, resulting in a better anti-noise ability. However, some nodes that are useful for prediction are excluded in the LPVG, resulting in the loss of information and inaccurate predictions. An example of LPVG algorithm ($\lambda = 1$) is given in Fig. 1c: Solid lines represent edges generated by the VG algorithm, and dashed lines represent new edges. The LPVG will degenerate to the VG when $\lambda = 0$ (no new edges in Fig. 1a).

The limited penetrable horizontal visibility graph (LPHVG) [92] is similar to LPVG. The difference is that there should be less than $\lambda$ intermediate data $(t_k, y_k)$ are

$$y_k > \inf \{ y_i, y_j \}.$$  \hspace{1cm} (6)

The LPHVG will also be equivalent to HVG when $\lambda = 0$ (no new edges).
The directed-limited penetrable horizontal visibility graph (DLPHVG) and the image-limited penetrable horizontal visibility graph (ILPHVG), which are named as the LPHVG family, have also been developed [93]. Numerical experiments have been taken to investigate topological properties of these networks, and they can be used to show the irreversibility of time series and discriminate noise from chaos. The sequential motifs in the LPVF can extract the hidden information in different flow patterns [94].

### 3.3 Multiscale limited penetrable horizontal visibility graph

In order to investigate the time series from the multiscale perspective, the multiscale limited penetrable horizontal visibility graph (MLPHVG) [95] has been proposed. The steps of this algorithm are shown below:

**Step (1)** Obtain the time series \((t_i, y_i)\), where \(t_i = 1, 2, \ldots, T\).

**Step (2)** Define the temporal scale and obtain the coarse-grained time series \(y_j^{[\kappa]}\),

\[
y_j^{[\kappa]} = \frac{\sum_{i=(j-1)\kappa+1}^{j\kappa} y_i}{\kappa}, \quad 1 \leq j \leq T/\kappa, \tag{7}
\]

where \(\kappa\) is the scale factor.

**Step (3)** The complex network is then constructed from the coarse-grained time series based on the LPHVG algorithm [92] with different scales \(\kappa\).

### 3.4 Multiplex visibility graph

Since physical, economic, and biological phenomena can be more effectively and accurately described by the multivariate time series, there have been numerous technologies and approaches to analyze the multidimensional signal. From the visibility graph perspective, the multiplex visibility graph (MVG) can map the multidimensional time series into multi-layer networks, thereby extracting information in the time series by analyzing the topological structural characteristics of the multi-layer network [96].

For a multivariate time series (\(M\)-dimension), \(x(t_i)\) is given as \(\{x^{[1]}(t_i), x^{[2]}(t_i), \ldots, x^{[M]}(t_i)\} \in \mathbb{R}^M\) for \(\forall 0 < t_i < T\). A \(M\)-layer multiplex network can be constructed, and each layer (e.g., layer \(\xi\)) can be constructed by the time series of \(x^{[\xi]}(t_i)\) based on the VG algorithm. It is important to note that other types of visibility graph can also be applied to define each layer of the multiplex graph. One 3-layer multiplex network under the undirected HVG is constructed in Fig. 3 as an example. For the \(M\)-layer multiplex network, the topological structure is defined by the vector of adjacency matrices \(\{A^{[1]}, A^{[2]}, \ldots, A^{[M]}\}\), where \(A^{[\xi]} = \{a_{ij}^{[\xi]}\}\) represent the adjacency matrix of layer \(\xi\), and \(a_{ij}^{[\xi]}\) shows whether node \(i\) and node \(j\) are connected in layer \(\xi\) [97]. Therefore, the information of the multivariate time series can be analyzed by the structure and characteristics of the multiplex network.

In order to investigate the information shared by different variables of the high-dimensional system, two
Fig. 2  The example of random and fractal time series with corresponding degree distribution of random and scale-free networks. The fractal time series is a Brownian motion with parameter $0.2$. Image generated based on Ref. [57]

(a) Random time series  
(b) Degree distribution of the random network

(c) Fractal time series  
(d) Degree distribution of the scale-free network

Fig. 3 Mapping a multivariate time series $\{x^{(\xi)}(t_i)\}_{\xi=1}^3$ into a 3-layer multiplex network by the MVG algorithm under the HVG geometrical criterion. Each layer ($L_\alpha$) of the multiplex network is constructed by $x^{(\alpha)}(t_i)$ in the multivariate time series by the HVG algorithm. Image generated based on Ref. [96]
measures are developed in the multiplex network. The average edge overlap is defined to show the abundance of single edges across layers,

\[
\alpha = \frac{\sum_i \sum_{j>i} \sum_{\xi} a_{ij}^{[\xi]}}{M \sum_i \sum_{j>i} \left(1 - \delta_{0, \sum_{\xi} a_{ij}^{[\xi]}}\right)},
\]

(8)

where \(\alpha \in [1/M, 1]\), and two cases are discussed below.

1. \(\alpha = 1/M\): Only \(a_{ij}^{[\xi]} = 1\), and \(a_{ij}^{[\xi]} = 0\) for \(\forall \xi \neq \xi'.\)
2. \(\alpha = 1\): The networks in \(M\) layers are the same.

The interlayer mutual information is then developed to investigate the interlayer correlation between the degree of the same node in different layers,

\[
I_{\xi,\zeta} = \sum_{k^{[\xi]}} \sum_{k^{[\zeta]}} \log \frac{P(k^{[\xi]}, k^{[\zeta]}) P(k^{[\xi]}, k^{[\zeta]})}{P(k^{[\xi]}) P(k^{[\zeta]})},
\]

(9)

where \(P(k^{[\xi]})\) and \(P(k^{[\zeta]})\) are the degree distribution in layer \(\xi\) and \(\zeta\), respectively, and \(P(k^{[\xi]}, k^{[\zeta]})\) represents the joint probability [96]. \(I = \{I_{\xi,\zeta}\}_{\xi,\zeta}\) can be then obtained to describe the typical amount of information flow.

Vector visibility graph (VVG) was also proposed to map multivariate time series into directed complex networks and study structures of stock markets [98]. The principled and systematic procedure that constructs the network from a multivariate time series was proposed by Kramer et al. [99], which was applied to analyze the signal during an epileptic seizure. The network can also be constructed by the small-shuffle surrogate approach from the multivariate time series [100], thereby showing the relationship of irregular fluctuations between two series. In order to study the nonlinear dynamic characteristic of gas–liquid flows, Gao et al. [101] have developed a new approach to map multivariate time series to a directed weighted complex network, which can investigate the transitions of different flow patterns. They then constructed a multiscale complex network based on the multivariate time series [102] and studied the coupled dynamical characteristics in the network by using a new clustering coefficient entropy in the gas–liquid two-phase flow experiment [103].

3.5 Others

The time complexity of the VG algorithm [57] is \(O(n^2)\), thus it needs a long time to map long time series to a complex network. In order to reduce the time complexity, Lan et al. [104] developed a fast transform VG algorithm based on the degree centrality of each node. They proposed the “divide & conquer” algorithm to find nodes from large degree to small degree and divide the time series into multiple segments to identify the connectivity between nodes, thereby achieving a shorter amount of time \(O(n \log n)\). Weighted complex networks can also be constructed from time series [105] based on the improved VG algorithm. The weight of links is determined by the induced ordered averaging aggregation operator and visibility graph aggregation operator, which can consider both the importance of nodes in the network and time decay in time series. Different dynamics in time series can construct different structures of the directed weighted network [106] because of orbits in the chaotic attractor.

The sequential visibility-graph motif [107] is a computationally efficient and highly informative feature, which can classify dynamics properties with noise, thereby being used to study the deterministic and stochastic dynamics in the time series. The parametric natural visibility graph (PNVG) algorithm will consider the view angle apart from the original features in VG [108], which can study the property that cannot be identified by the VG. When there is a window slide moving in the time series, the time-dependent limited penetrable visibility graph (TDLPVG) algorithm has been developed, which can analyze the statistical properties of the time series [109]. The parametric modified limited penetrable visibility graph (PMLPVG) is then improved based on the LPVG [110], so that it shows the dynamic properties of time series and enhances the reasonability of the penetrable visibility, which has been tested in different real-world time series. Due to the unavoidability of noise in time series, the improved power of scale-freeness of VG (PSVG) is developed to measure the fractality of time series by analyzing the fractality in different scales [111]. The time series can also be analyzed based on the recurrence network [112–114]. There are still many other types of visibility graphs with different features and geometrical criteria, which can be used to analyze different time series.

3.6 Application

Due to the effectiveness of the VG algorithm, the family of VG algorithms has been applied in different fields
to address practical problems, such as financial time series [115–117], electroencephalogram (EEG) signal [70, 118], traffic data [119, 120], earthquake time series [121, 122], and market dataset [123, 124].

In order to make the time series smoothing, the degree distribution in the VG that shows the importance of nodes is applied to determine the weight of aggregation, and the visibility graph averaging (VGA) aggregation operator is developed to aggregate several time series [125]. Inspired by the VGA aggregation operator, the visibility graph power averaging aggregation operator [126] is then proposed based on a new support function that takes into account the relationship and correlation between values. The ordered visibility graph weighted averaging aggregation operator [127] that takes into account the information of order and argument values is then developed to determine the weight to aggregate time series effectively. Xu et al. [128] linearly combined weights obtained by the visibility graph averaging operator and induced ordered weighted aggregation operator, thereby smoothing time series and avoiding information loss. The visibility graph averaging aggregation operator has also been used on the topic of dynamic data fusion, which is a data-driven approach to consider past data credibility without subjective parameters setting [73]. The HVG can be applied to analyze chaotic signals polluted with white noise [65], thereby showing the irreversibility of series with noise. The noise from chaos can also be distinguished by the HVG algorithm [129]. The nonstationary time series can also be analyzed by the variant of VG algorithm [109, 130]. Through the application in the exchange rate series, the original, shuffled, and detrended series can be analyzed by the VG algorithm [63].

The visibility graph algorithm is also applied to study the feature of EEG signals, including the diagnosis of Alzheimer’s disease [131], sleep stages classification [118], epileptic seizure detection [132, 133], and alcoholism identification [134]. Apart from the EEG signal, the seismicity sequences can also be analyzed by the VG [135], and they found the power-law distribution of magnitude intensity. In addition, the time series of energy dissipation rates in turbulence [136] can be investigated by the visibility graph. The statistical characteristic of the network is then studied by the degree distribution and the box-covering algorithm. The fractional Brownian motions and multifractal random walks are then analyzed by the visibility graph [137], and they found the degree of nodes in the VG follows power-law distributions, which is influenced by the temporal correlation of the time series. Lacasa et al. [62] also mapped the fractional Brownian motion into a scale-free network to study the relationship between the degree distribution and Hurst parameter $H$, thereby investigating long-range dependence in this kind of time series. The synchronization between time series has been measured by the visibility graph similarity (VGS) approach [138]. The VGS indicates the correlation between the degree sequence of different systems, thereby showing the similarity of dynamics between two coupled systems (temporal synchronization).

4 Time series prediction

In order to forecast the time series, many algorithms have been proposed. For example, the linear autoregressive model was proposed based on a linear function [139]. The local linear predictor [140] is then developed which can achieve a similar performance to the global linear predictor. A transformer-based model [141] was proposed to predict long sequence time-series, which has a better performance than existing methods. Several time series prediction methods based on deep learning [142] were reviewed. We will also review the time series prediction models based on the visibility graph algorithm. This method analyzes the property of time series from the structural information of the complex network, which is a new perspective. To accurately forecast the time series from the visibility graph perspective, the similarity between nodes needs to be evaluated first to develop the prediction model. In the complex network, the similarity between nodes can be evaluated by numerous approaches [75], including maximum likelihood models, probabilistic methods, and structure-based algorithms.

Zhang et al. [74] first developed a model based on the visibility graph and local random walk in the network to forecast time series. Given a time series $(t_i, y_i)$, where $t_i = 1, 2, \ldots, T$, the network is constructed by the VG algorithm [57]. The topological structure can be represented by the adjacency matrix $A_{|N| \times |N|}$, where $a_{ij}$ shows whether node $i$ is connected to node $j$. $k_i = \sum_j a_{ij}$ is the degree of node $i$. Note that $T$ in the time series equals to $|N|$ in the network. The similarity between the last node at time $T$ (corresponding to $(t_T, y_T)$) and other nodes is evaluated by the local
random walk approach [77]. Specifically, the transition probability that a random walker at node \( i \) walks to node \( j \) each step is represented by,

\[
P_{ij} = \frac{a_{ij}}{k_i},
\]

where \( P_{ij} \) is an element in the probability matrix \( P_{ij \times N} \). Hence, the probability that a walker reaches node \( j \) from node \( i \) after \( t \) steps is defined by,

\[
\pi_i(t) = P^T \pi_i(t-1),
\]

where \( \pi_i \) is an all-zero vector \( (|N| \times 1) \) except the \( i \)th element is 1. The similarity between nodes based on the local random walk approach is then obtained by,

\[
S_{ij}(t) = \frac{k_i}{2|E|} \pi_{ij}(t) + \frac{k_j}{2|E|} \pi_{ji}(t).
\]

Note that \( t \) is the time that the walker needs in the network, which is not related to the time series. In addition, \( S_{ij} \) is a symmetric matrix \( (S_{ij} = S_{ji}) \). In order to prevent this walker from going away from both node \( i \) and node \( j \), the random walker should walk in the local network rather than the entire network, resulting in a higher similarity,

\[
S_{ij} = \sum_{l=1}^{t} S_{ij}(l).
\]

After identifying the most similar node \( (t_s, y_s) \) to the last node \( (t_T, y_T) \) at time \( T \), the time series \( (t_{T+1}^{*}, y_{T+1}^{*}) \) at the next time \( T + 1 \) is obtained by,

\[
\hat{y}_{T+1} = \frac{y_T - y_s}{t_T - t_s} (t_{T+1}^{*} - t_T) + y_T.
\]

The framework of this method is shown in Fig. 4a. When the value at \( T + 1 \) is predicted, a new network with \( T + 1 \) nodes will be constructed by the VG algorithm, thereby predicting the value at the next time \( T + 2 \). This is called the one-step-ahead prediction because it only predicts the value at the next time. A multi-step-ahead prediction approach has also been proposed to increase the applicability of the model [74]. This model was further improved by fuzzy logic [78]. After identifying the most similar node \( (t_s, y_s) \) to the last node \( (t_T, y_T) \), the distance between two nodes is determined by,

\[
d_{ST} = \left| t_T - t_s \right|,
\]

and the fuzzy variable is obtained by,

\[
r = \log_2 d_{ST}.
\]

Three factors are then considered to design the fuzzy rule in this method, and they are:

1. **Range of visibility** Some old but still visible data is not conducive to the current prediction.
2. **Level of visibility** Several fuzzy sets are used to describe the level of visibility.
3. **Effect of visibility** The predicted data \( (t_{T+1}^{*}, y_{T+1}^{*}) \) is determined by (a) the direct effect from \( (t_T, y_T) \) and (b) the indirect effect from the initial forecasting result \( (t_{T+1}^*, \hat{y}_{T+1}) \).

Therefore, the weighting parameters of the direct effect \( w_1 \) and indirect effect \( w_2 \) are determined by the fuzzy rule,

\[
w_1, w_2 = \frac{\sum_{l=1}^{M} Q^l \mu_{pl}(r)}{\sum_{l=1}^{M} \mu_{pl}(r)}.
\]

The specific introduction about the fuzzy rule can be found in Ref. [78]. Finally, the predicted value can be obtained by two kinds of effect,

\[
\hat{y}_{T+1} = w_1 \times y_T + w_2 \times \hat{y}_{T+1}.
\]

The predicted value \( \hat{y}_{T+1} \) obtained by considering two kinds of information can give a more accurate prediction, and this method has high flexibility and predictability. The framework of this method is shown in Fig. 4b.

When \( (t_s, y_s) \) is farther away from \( (t_T, y_T) \), it will reveal more historical information of the time series, allowing this point to be more important. Hence, a new approach has been proposed [79] that determines the weighting parameters by the distance between nodes,

\[
w_1 = \frac{d_T(T+1)}{d_S(T+1)},
\]

\[
w_2 = \frac{d_{ST}}{d_S(T+1)}.
\]

The predicted value \( \hat{y}_{T+1} \) is also obtained by Eq. (18) with new weighting parameters.

Mao and Xiao [80] then developed a model that combines the linear approximation model and the adjacent prediction approach. Specifically, the impact of \( (t_T, y_T) \) on \( (t_{T+1}, y_{T+1}) \) is similar to the impact of \( (t_s, y_s) \) on \( (t_{S+1}, y_{S+1}) \) due to the similarity between \( (t_T, y_T) \) and \( (t_s, y_s) \), and thus, the adjacent prediction approach is defined by,

\[
\hat{y}_{T+1} = \frac{y_s}{t_{S+1} - t_s} (t_{T+1} - t_T) + y_T.
\]

The linear approximation prediction model is given in Eq. (14), and the weighting parameters are obtained by
the distance between nodes (Eq. (19)). Therefore, the final prediction result \( \hat{y}_{T+1} \) is obtained by,

\[
\hat{y}_{T+1} = w_1 \times \hat{y}_T + w_2 \times \tilde{y}_{T+1}.
\]  

(21)

The framework of this method is shown in Fig. 4c.

After constructing the complex network from the fast transform algorithm [104], Zhao et al. [81] proposed a method to consider the impact of all nodes that are visible to the last node \((t_T, y_T)\). The set of nodes that are visible to \((t_T, y_T)\) is denoted by \( t^* = \{t_i^*\}, i \in R \). If there is only one visible node in \( t^* \), the initial predicted value on the visible node \( t_S = t_i^* \) can be obtained by Eq. (20), and the revised predictive value is obtained by,

\[
y_T + \tilde{y}_{T+1} \]

\[
\hat{y}_{T+1} = \frac{y_T + \tilde{y}_{T+1}}{2}.
\]  

(22)

However, the time series is usually complicated. Therefore, there are more than one visible nodes in \( t^* \) in general. Hence, the effect of all visible nodes on future nodes should be considered, and the final predicted value \( \hat{y}_{T+1} \) is defined by,

\[
\hat{y}_{T+1} = \sum_{i^*} w_{[i^*]} y_{T+1}^{[i^*]},
\]  

(23)

where \( w_{[i^*]} \) is the weighting parameter of the visible node at \( t_i^* \) that is defined by,

\[
w_{[i^*]} = \frac{k_{i^*}/d_{i^*}}{\sum_{i^*} k_{i^*}/d_{i^*}.}
\]  

(24)

where \( k_{i^*} \) is the degree of node, \( d_{i^*} \) is the distance obtained by Eq. (15). Therefore, this approach considers more historical information in the time series, resulting in a comprehensive prediction. Recently, Hu and Xiao [143] have found that the directed VG algorithm still perform well after adding observational noise.

Local random walk has been applied in these methods to predict the potential link, but it leads to a high time complexity. A fast algorithm based on Markov chain was then developed to forecast the time series [144]. The similarity between nodes is obtained by the stationary distribution of the probability transfer matrix,

\[
V_{TI}^x(i) = P^x V_0,
\]  

(25)

where \( V_0 \) is the probability distribution of the initial nodes, \( P \) is the transfer matrix, and \( V_{TI}^x(i) \) describes the similarity between the last node \((t_T, y_T)\) and other nodes \( i \). This method needs \( x \) iteration for \(|V^{x+1} - V^x| < 10^{-5}\). In addition, this method considers more historical information of the time series, and top \( k \) most similar nodes (the top \( k \) maximum value in \( V \)) are then selected to forecast the time series,

\[
\hat{y}_{T+1} = \sum_{S=1}^{k} \frac{V_S}{k} \left( y_T - y_S (t_{T+1} - t_T) + y_T \right),
\]  

(26)

where weighting parameters are determined by the similarity degree. The value of \( k \) is determined by the trend of the time series:

(1) \( k = 2 \): The time series has a downward or upward trend.

(2) \( k = 4 \): It is a periodic time series.

Liu et al. [82] then developed a fuzzy interval model to forecast hydrological and financial time series. In their method, the time series needs to be decomposed and reconstructed first, the predicted values in different time-frequency spaces are then aggregated to obtain the final result. The steps of this approach are shown below:

(1) The time series \((t_i, y_i) (i = 1, 2, \ldots, T)\) is decomposed into different intrinsic mode functions (IMFs),

\[
y_i = \sum_{j=1}^{n} c_i^{[j]} + r_i^{[n]},
\]  

where \( r_i^{[n]} \) is the residue after \( T \) IMFs are extracted. The detail settings can be found in Ref. [82].

(2) The time series can be then reconstructed by the residue of different time-frequency spaces.

(3) For each reconstructed part, the time series will be used to construct the VG (Eq. (2)), and the similarity between nodes can be measured by the local random walk approach (Eqs. (12) and (13)).

(4) The predicted value for each reconstructed time series can be obtained by Eq. (14).

(5) The golden-rule-based representation function [145] is then used to determine the membership of the forecasting interval for each time-frequency space.

(6) The final predicted value can be obtained by aggregating the values in different time-frequency space via the induced ordered weighted averaging aggregation operation [146].
The Construction Cost Index (CCI)\(^1\) describes the average price of several factors in the construction industry from January 1990 to July 2014, rendering people to estimate costs and make budgets in advance, thus attracting researchers’ interests. These models based on the visibility graph are used to forecast the CCI dataset. Using the approach in Ref. [74], predicted values and actual values of CCI data are shown in Fig. 5a. In the entire time span, predicted values are similar to real values, thereby showing the performance of VG-based forecasting models. The constructed VGs in different steps (\(T = 50\), \(T = 100\), and \(T = 200\)) are shown in Fig. 5b–d, respectively. The last node at time \(T\) and its most similar node are represented by the red node and blue node, respectively. Several criteria are then used to evaluate the performance of

\(^1\) Engineering News-Record: http://enr.construction.com/economics/.
the predictability, including mean absolute difference (MAD), mean absolute percentage error (MAPE), root mean square error (RMSE), and normalized root mean squared error (NRMSE),

\[
\begin{align*}
\text{MAD} &= \frac{1}{n} \sum_{i=1}^{n} |\hat{y}_i - y_i|, \\
\text{MAPE} &= \frac{1}{n} \sum_{i=1}^{n} \frac{|\hat{y}_i - y_i|}{y_i} \times 100, \\
\text{RMSE} &= \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}, \\
\text{NRMSE} &= \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2} \left( \frac{y_{\text{max}} - y_{\text{min}}}{100} \right)
\end{align*}
\]

where \(y_i\) is the actual value, \(\hat{y}_i\) is the predicted value, \(n\) is the number of data. The simple moving average (\(k = 1\)) [147] is applied as the comparison method. The results of the comparison method and several VG-based approaches (in-sample forecasting) are given in Table 1. As observed from the results, these VG-based models can better forecast CCI time series with lower errors.

A sliding window over data has also been applied to validate the forecasting via the out-of-sample prediction. The algorithm of the out-of-sample prediction is shown in Algorithm 2.

Two approaches are used to make out-of-sample predictions, and the results with different lengths of the sliding window \(L\) are shown in Table 2. When the length of the sliding window \(L\) becomes longer, the MAPE is also larger. In addition, the MAPE obtained by the approach in Ref. [144] is lower than the MAPE obtained by the model in Ref. [78]. The average values of MAPE under \(L = 3, 6, 12\) are then used to evaluate the performance of different methods. Two VG-based approaches are compared with Holt ES, Holt-Winter ES, ARIMA, Seasonal ARIMA, and official ENR data. The results (Table 3) show that the VG-based approaches achieve the lowest MAPE and have better performance.

### Table 1 Errors of different approaches in CCI dataset (in-sample forecasting)

| Reference | MAD   | MAPE(%) | RMSE | NRMSE(%) |
|-----------|-------|---------|------|----------|
| Ref. [147]| 21.59 | 0.3110  | 32.73| 0.6350   |
| Ref. [74] | 20.05 | 0.2900  | –    | –        |
| Ref. [78] |       |         |      |          |
| Phase 1   | 20.0  | 0.2889  | 29.3 | 0.5690   |
| Phase 2   | 19.6  | 0.2845  | 29.1 | 0.5659   |
| Ref. [79] | 19.30 | 0.2797  | 28.16| 0.5462   |
| Ref. [81] | 19.92 | 0.2882  | 29.06| 0.5600   |
| Ref. [144]| 19.88 | 0.2899  | 28.06| 0.5455   |

Data obtained from Refs. [74,78,79,81,144]

### Table 2 MAPE of two VG-based prediction models with different values of \(L\) in CCI dataset (out-of-sample forecasting)

| Reference   | \(L = 3\) (%) | \(L = 6\) (%) | \(L = 12\) (%) |
|-------------|---------------|---------------|----------------|
| Zhang et al. [78] | 0.48          | 0.75          | 1.27           |
| Liu et al. [144]  | 0.45          | 0.62          | 0.84           |

Data obtained from Refs. [78,144]

### Algorithm 2 Out-of-sample forecasting with a sliding window.

**Input:** Data set \((i, y_i)\); Size of data set \(T\); Length of the sliding window \(L\).

**Output:** Forecasting value \(\hat{y}_i\); \(\text{MAPE}\);

1. for \(i = 1\) to \(T - L + 1\) do
2. for \(j = 1\) to \(L\) do
3. \(\hat{y}_{i+j} = f(y_i, y_{i+1}, \ldots, y_{i+j})\); \(\triangleright f(\bullet)\) is the forecasting model.
4. \(e(j) = \text{APE}(\hat{y}_{i+j}, y_{i+j+1})\); \(\triangleright \text{APE}\) is the absolute percentage error.
5. end for
6. \(\text{MAPE}(i) = \text{Average}(e); \)
7. end for
8. \(\text{MAPE} = \text{Average}(\text{MAPE}(i)); \triangleright \text{MAPE}\) is the mean absolute percentage error.

### 5 Image classification

Apart from one-dimensional time series, the family of VG algorithms can also be applied to two-dimensional images [83], including image processing and classification. In the network, each node represents a pixel in the image, and the connection between nodes is derived from the image visibility graph (IVG) algorithm. Specifically, a \(N \times N\) image is represented by its pixel matrix \(I_{N \times N} \in R\), and there will be \(N^2\) nodes in corresponding IVG. Followed by the IVG algorithm, two nodes \(ij\) and \(ij'\) will be connected if they fulfill both conditions:

\(i = i'\) or \(j = j'\) or \((i = i' + p) \wedge (j = j' + p)\)

for any integer \(p\).
Table 3 The average value of MAPE under \( L = 3, 6, 12 \) obtained by different approaches in CCI dataset (out-of-sample forecasting)

| Method          | Holt ES | Holt-Winter ES | ARIMA  | Seasonal ARIMA | ENR    | Zhang et al. [78] | Liu et al. [144] |
|-----------------|--------|---------------|--------|----------------|--------|-------------------|------------------|
| MAPE            | 1.03%  | 1.00%         | 1.29%  | 1.06%          | 1.22%  | 0.83%             | 0.64%            |

Data obtained from Refs. [78,144]

(2) \( I_{ij} \) and \( I_{i'j'} \) are connected in the VG.

If the second condition is that \( I_{ij} \) and \( I_{i'j'} \) are connected in the HVG, the constructed network will be the image horizontal visibility graph (IHVG). Connections in both IVG and IHVG have specific directions (rows, columns, and diagonals) in the image according to the first condition. When the pixel matrix \( I_{3\times3} \) is given as

\[
I_{3\times3} = \begin{bmatrix}
105 & 90 & 100 \\
50 & 95 & 90 \\
90 & 10 & 40 \\
\end{bmatrix},
\]

the constructed IHVGs with and without the lattice structure are shown in Fig. 6.

The family of IVG algorithms can be applied as image filters. For a given image \( I_{N\times N} \), the visibility filter obtained by the IVG and IHVG are defined as \( F_{VG}(I)_{N\times N} \) and \( F_{HVG}(I)_{N\times N} \), respectively. Filters \( F_{VG}(I)_{ij} \) and \( F_{HVG}(I)_{ij} \) are defined by the property of node in the constructed network, such as \( k \)-filter based on the degree, \( C \)-filter based on the clustering coefficient, and \( K_{nn} \)-filter based on the degree–degree [148]. Specifically, the clustering coefficient needs to be rescaled in grayscale intensity \([0, 255]\) in the \( C \)-filter. Several standard grayscale images and their IVG \( k \)-filter are shown in Fig. 7.

More importantly, the family of IVG algorithms can be used in pattern recognition and texture classification based on three kinds of features.

(1) Multiplex features It is usually applied to deal with several images at once, such as RBG images. These images will construct an image multiplex visibility graph (IMVG). The features in the multilayer network can be described by three types of descriptors, including the intra-layer descriptors, inter-layer descriptors, and intrinsically multiplex descriptors. This kind of feature needs further exploration because multilayer networks are still not widely used in images.

(2) Global features It is defined by the topological structural information of the entire network, such as the degree distribution and clustering distribution. Other statistical characteristics of networks can also be applied here.

(3) Local features It is mainly defined by the visibility patches (VPs) in the network, with order \( p \), the visibility patch contains \( p^2 \) nodes \( \{ij\}_{l=r1,l=r2}^{r1+p-1,r2+p-1} \) for \( 1 \leq r1 \leq N - p \) and \( 1 \leq r2 \leq N - p \). In the network, the lowest order is \( p = 3 \) because the structure of all VPs will be the same when the order is \( p = 2 \). The VP with order \( p \) can be represented by a \( 2p + 2 + q \)-dimensional vector,

\[
[r_1, r_2, \ldots, r_p, c_1, c_2, \ldots, c_p, d_1, d_2, l_1, l_2, \ldots, l_q],
\]

where \( q = 4(p-2) \). Specifically, there are \( p \) rows, \( p \) columns, main diagonal and anti-diagonal of the patch, and \( q \) motifs of lower orders of off-diagonals and off-anti-diagonals.

An example of VP with order \( p = 3 \) and its label are shown in Fig. 8. When two outer nodes are connected, the label is 1, otherwise the label is 0 (Fig. 8). There are total \( 2^{2(p+2)} = 2^8 = 256 \) kinds of patches with order \( p = 3 \) in the network. The vector of the patch shown

\[ \text{Image filters. For a given image} \]
in Fig. 8 is \[ 1, 0, 0, 1, 0, 0, 1, 0 \], and thus, the label can be obtained by,
\[
L = \sum_{i=1}^{2p+2+q} V_i 2^{2p+2+q-i} \sum_{i=1}^{8} V_i 2^{8-i} = r_1 2^7 + \ldots + d_2 2^0 + 1 = 147.
\]
Hence, VP can be used to represent the local feature of the network, and the number of VP with different labels (VP distribution) can be further used to classify images with different properties.

After collecting these three types of features of images, the principal component analysis (PCA) is used to reduce the number of features, thereby avoiding overfitting. Different kinds of support vector machine (SVM) algorithms are then applied to classify images into different categories.

The colored Brodatz dataset [151] has 12 classes and is widely applied to analyze texture and understand scanned images. Several IVG-based approaches and comparison methods based on artificial intelligence are used to classify images into different categories in this dataset (Table 4). The global and local multiplex features in the IVG and IHVG are both used to represent and classify images [83], the best classification accuracy obtained from all classifiers in the MATLAB Classification Learner is 99.8%. The texture classification based on image visibility graph (TCIVG) [84] considers the degree distribution of IVG and IHVG to classify images, and the best classification accuracy of this approach (91.4%) is obtained by applying quadratic discriminant in the IVG without lattice. Images are then divided into multiple segments by the matching intensity for image visibility graphs (MIVG) [85], and the structure is described by the reference patterns and matching intensity. The best classification accuracy of this approach is 99.7%, and it is obtained by linear discriminant in MIVG-1 and quadratic discriminant in MIVG-2. The best classification accuracy of three comparison methods (CNN, VGG 16, and VGG19) in this dataset is 99.2%, 99.6%, and 99.7%, respectively. Therefore, IVG-based algorithms outcompete other comparison methods based on artificial intelligence on image classification.

---

**Table 4** The best classification accuracy obtained by different IVG-based models and comparison methods based on artificial intelligence in the colored Brodatz dataset

| Method         | Classifier                      | Accuracy (%) |
|----------------|---------------------------------|--------------|
| IVG & IHVG [83]| Support vector machine         | 99.8         |
| TCIVG [84]     | Quadratic discriminant          | 91.4         |
| MIVG [85]      | Linear & quadratic discriminant | 99.7         |
| CNN [149]      | Support vector machine          | 99.2         |
| VGG16 [150]    | Linear discriminant             | 99.6         |
| VGG19 [150]    | Linear discriminant             | 99.7         |

Data obtained from [83–85]
IVG has also been applied as a parameter-free model to explore the local and global structure of signatures, thereby verifying signatures [152]. This approach can achieve higher accuracy than other classic state-of-the-art models. Pessa et al. [153] then proposed an approach to map images to ordinal networks by determining the structure of networks from the symbolization process. They have shown that this model can be used to identify image symmetries and classify textures in real-world datasets. In addition, their method is robust against noise, resulting in higher classification accuracy.

6 Conclusions and future outlooks

Analyzing time series and images from the perspective of complex networks is an innovative and effective method to address interdisciplinary problems. The characteristic of time series and images, including non-linear dynamics, chaos, irreversibility, and local information of images, can be analyzed through the topological structure of complex networks. After combining artificial intelligence and information fusion models, the visibility graph can be applied to forecast time series and classify images, which can achieve a good performance. In this review paper, the visibility graph that can map time series and images to complex networks is discussed. Different kinds of rules that construct complex networks from time series are introduced in Sect. 2, where the visibility graph is the focus of this review paper. Hence, the family of visibility graphs is then introduced in Sect. 3 to construct different types of networks, including (un-) weighted, (un-) directed, and (single-) multi-layered networks. Specifically, the application of visibility graph algorithms in different fields is discussed in Sect. 3.6. The visibility graph can be combined with the local random walk algorithm and other information fusion models to forecast financial time series with in-sample forecasting and out-of-sample forecasting (Sect. 4). Through the comparison in the CCI dataset, the VG-based models are shown to achieve a better performance. The image visibility graph algorithm is then introduced in Sect. 5 to map images to complex networks, and images can be classified by considering three types of features extracted from the image visibility graph with the help of the support vector machine and other machine learning models. By considering the multiplex, global, and local features in the complex network, the IVG-based algorithms outcompete other comparison methods based on artificial intelligence in the colored Brodatz dataset.

We now highlight several research directions in this field that are of theoretical and practical value:

(1) The family of visibility graph algorithms has been applied to many fields to solve various applied and engineering problems (as discussed in our main text). However, researchers have not developed corresponding visibility graph algorithms based on intrinsic features (such as period, dynamic characteristics, and local information) of different types of time series (including finance, earthquake, and EEG signal), but still use the same algorithm to solve problems from various disciplines. That is to say, existing visibility graph algorithms are not developed and improved based on the background of the problem statement. Therefore, algorithms for constructing complex networks based on the feature of time series still need further exploration to discover time series characteristics more accurately.

(2) In complex networks, there are many algorithms to measure the similarity between nodes, such as common neighbors, SimRank, and superposed random walk. However, only the local random walk model is used in the existing time series prediction models. Can other models that consider more structural information about the time series be proposed...
to forecast the time series then? How can weights in these models be determined more reasonably?

(3) The visibility graph algorithms have been proposed to address problems on the topic of time series and images. Can it be further extended to solve audio and video data to increase the scope of application of the visibility graph? Since the network-based analysis model can describe complicated relationships between data, it may provide a new perspective for analyzing the ever-increasing high-dimensional and large-scale datasets.

(4) The visibility graph algorithms can be more closely integrated with artificial intelligence models. The data-driven machine learning model can mine the universal laws and features hidden behind the data. The visibility graph algorithm studies the visibility between the data through the network, thereby providing a new research perspective. Hybrid models will not only improve the accuracy of time series prediction and image classification, but also reduce the dimensionality of complicated time series and images dataset, thus revealing more features.

Acknowledgements This project was supported by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 2 (Grant No. MOET2EP50120-00021).

Data availability All data generated or analyzed during this study are included in this published article.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Marx, V.: The big challenges of big data. Nature 498(7453), 255–260 (2013)
2. Fan, J., Han, F., Liu, H.: Challenges of big data analysis. Natl. Sci. Rev. 1(2), 293–314 (2014)
3. LaValle, S., Lesser, E., Shockley, R., Hopkins, M.S., Kruschwitz, N.: Big data, analytics and the path from insights to value. MIT Sloan Manag. Rev. 52(2), 21–32 (2011)
4. Xiao, F.: CEQD: a complex mass function to predict interference effects. IEEE Trans. Cybern. 52(8), 7402–7414 (2022)
5. Hamilton, J.D.: Time Series Analysis. Princeton University Press (2020)
6. Haralick, R.M., Shanmugam, K., Dinstein, I.H.: Textural features for image classification. IEEE Trans. Syst. Man Cybern. 6, 610–621 (1973)
7. Huang, Y., Mao, X., Deng, Y.: Natural visibility encoding for time series and its application in stock trend prediction. Knowl.-Based Syst. 232, 107478 (2021)
8. Enders, W.: Applied Econometric Time Series. Wiley (2008)
9. Liu, H., Song, W., Zio, E.: Generalized cauchy difference iterative forecasting model for wind speed based on fractal time series. Nonlinear Dyn. 103(1), 759–773 (2021)
10. Affonso, C., Rossi, A.L.D., Vieira, F.H.A., de Leon Ferreira de Carvalho, A.C.P.: Deep learning for biological image classification. Expert Syst. Appl. 85, 114–122 (2017)
11. Chen, H.-C., Wei, D.-Q.: Chaotic time series prediction using echo state network based on selective opposition grey wolf optimizer. Nonlinear Dyn. 104(4), 3925–3935 (2021)
12. Wang, J., Yan, Z., Gui, L., Xu, K., Lan, Y.: Reconstruction of nonlinear flows from noisy time series. Nonlinear Dyn. 108, 3887–3902 (2022)
13. Sezer, A., Altan, A.: Detection of solder paste defects with an optimization-based deep learning model using image processing techniques. Solder. Surf. Mount Technol. (2021)
14. Karasu, S., Altan, A.: Crude oil time series prediction model based on LSTM network with chaotic Henry gas solubility optimization. Energy 242, 122964 (2022)
15. Altan, A., Karasu, S., Zio, E.: A new hybrid model for wind speed forecasting combining long short-term memory neural network, decomposition methods and grey wolf optimizer. Appl. Soft Comput. 100, 106996 (2021)
16. Yu, Z., Abdel-Salam, A.-S.G., Sohail, A., Alam, F.: Forecasting the impact of environmental stresses on the frequent waves of COVID19. Nonlinear Dyn. 106(2), 1509–1523 (2021)
17. Liu, Y., Gong, C., Yang, L., Chen, Y.: DSTP-RNN: a dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction. Expert Syst. Appl. 143, 113082 (2020)
18. Li, J., Shang, P., Zhang, X.: Time series irreversibility analysis using Jensen-Shannon divergence calculated by permutation pattern. Nonlinear Dyn. 96(4), 2637–2652 (2019)
19. Chen, Y., Lin, A.: Weighted link entropy and multiscale weighted link entropy for complex time series. Nonlinear Dyn. 105(1), 541–554 (2021)
20. Altan, A., Karasu, S.: Recognition of COVID-19 disease from X-ray images by hybrid model consisting of 2D curvelet transform, chaotic salp swarm algorithm and deep learning technique. Chaos Solitons Fractals 140, 110071 (2020)
21. Barabási, A.-L., Gulbahce, N., Loscalzo, J.: Network medicine: a network-based approach to human disease. Nat. Rev. Genet. 12(1), 56–68 (2011)
22. Mantegna, R.N., Stanley, H.E.: Scaling behaviour in the dynamics of an economic index. Nature 376(6535), 46–49 (1995)
23. Wen, T., Cheong, K.H.: The fractal dimension of complex networks: a review. Inf. Fusion 73, 87–102 (2021)
24. Newman, M.: Networks. Oxford University Press (2018)
25. Wen, T., Deng, Y.: Identification of influencers in complex networks by local information dimensionality. Inf. Sci. 512, 549–562 (2020)
26. Wang, L., Ma, L., Wang, C., Xie, N.-G., Koh, J.M., Cheong, K.H.: Identifying influential spreaders in social networks
through discrete moth-flame optimization. IEEE Trans. Evol. Comput. 25, 1091–102 (2021)

27. Wen, T., Gao, Q., Kalmár-Nagy, T., Deng, Y., Cheong, K.H.: A review of predator-prey systems with dormancy of predators. Nonlinear Dyn. 107, 3271–3289 (2022)

28. Tan, Z.X., Cheong, K.H.: Nomadic-colonial life strategies enable paradoxical survival and growth despite habitat destruction. Else if 6, e21673 (2017)

29. Cheong, K.H., Wen, T., Benler, S., Koh, J.M., Koonin, E.V.: Alternating lysis and lysisogeny is a winning strategy in bacteriophages due to Parrondo’s paradox. Proc. Natl. Acad. Sci. 119(13), e2115145119 (2022)

30. Wen, T., Cheong, K.H., Lai, J.W., Koh, J.M., Koonin, E.V.: Extending the lifespan of multicellular organisms via periodic and stochastic intercellular competition. Phys. Rev. Lett. (2022)

31. Zhao, Y., Liu, P.X., Wang, H., Bao, J.: Funnel-bounded synchronization control for bilateral teleoperation with asymmetric communication delays. Nonlinear Dyn. 107, 3641–3654 (2022)

32. Wang, C., Ji, J., Miao, Z., Zhou, J.: Synchronization control for networked mobile robot systems based on Udwadia-Kalaba approach. Nonlinear Dyn. 105(1), 315–330 (2021)

33. Cheong, K.H., Wen, T., Lai, J.W.: Relieving cost of epidemic by Parrondo’s paradox: a COVID-19 case study. Adv. Sci. 7(24), 2002324 (2020)

34. Wen, T., Cao, J., Cheong, K.H.: Gravity-based community vulnerability evaluation model in social networks: Gbcve. IEEE Trans. Cybern. (2021). https://doi.org/10.1109/TCYB.2021.3123081

35. Wang, C., Tan, Z.X., Ye, Y., Wang, L., Cheong, K.H., Xie, N.-G.: A rumor spreading model based on information entropy. Sci. Rep. 7(1), 1–14 (2017)

36. Wen, T., Deng, Y.: The vulnerability of communities in complex networks: an entropy approach. Reliab. Eng. Syst. Saf. 196, 106782 (2020)

37. Pan, L., Gao, X., Deng, Y., Cheong, K.H.: The constrained pythagorean fuzzy sets and its similarity measure. IEEE Trans. Fuzzy Syst. 30, 1102–13 (2021)

38. Xiao, F.: CaFiR: a fuzzy complex event processing method. Int. J. Fuzzy Syst. (2021). https://doi.org/10.1007/s40815-021-01118-6

39. Ye, Y., Hang, X.R., Koh, J.M., Miszczak, J.A., Cheong, K.H., Xie, N.-G.: Passive network evolution promotes group welfare in complex networks. Chaos Solitons Fractals 130, 109464 (2020)

40. Wen, T., Koonin, E.V., Cheong, K.H.: An alternating active-dormitive strategy enables disadvantaged prey to outcompete the perennially active prey through Parrondo’s paradox. BMC Biol. 19(1), 168 (2021)

41. Fan, C., Zeng, L., Sun, Y., Liu, Y.-Y.: Finding key players in complex networks through deep reinforcement learning. Nat. Mach. Intell. 2(6), 317–324 (2020)

42. Liu, G., Xiao, F.: Time series data fusion based on evidence theory and OWA operator. Sensors 19(5), 1171 (2019)

43. Gao, Q., Wen, T., Deng, Y.: Information volume fractal dimension. Fractals 29(8), 2150263 (2021)

44. Zhu, W., Xiao, F.: Improvement of time series data fusion based on evidence theory and DEMATEL. IEEE Access 7, 81397–81406 (2019)

45. Lai, J.W., Chang, J., Ang, L.K., Cheong, K.H.: Multi-level information fusion to alleviate network congestion. Inf. Fusion 63, 248–255 (2020)

46. Chen, L., Deng, Y., Cheong, K.H.: Probability transformation of mass function: a weighted network method based on the ordered visibility graph. Eng. Appl. Artif. Intell. 105, 104438 (2021)

47. Wen, T., Gao, Q., Chen, Y.-W., Cheong, K.H.: Exploring the vulnerability of transportation networks by entropy: a case study of Asia-Europe maritime transportation network. Reliab. Eng. Syst. Saf. 226, 108578 (2022)

48. Deng, Y.: Random permutation set. Int. J. Comput. Commun. Control 17(1), 4542 (2022)

49. Zhou, Q., Deng, Y.: Belief eXtropy: measure uncertainty from negation. Commun. Stat. Theory Methods (2021). https://doi.org/10.1080/03610926.2021.1980049

50. Deng, Y.: Uncertainty measure in evidence theory. Sci. China Inf. Sci. 63(1), 210201 (2020)

51. Song, Y., Deng, Y.: Entropic explanation of power set. Int. J. Comput. Commun. Control 16(4), 4413 (2021)

52. Qiang, C., Deng, Y., Cheong, K.H.: Information fractal dimension of mass function. Fractals (2022)

53. Backes, A.R., Casanova, D., Bruno, O.M.: Texture analysis and classification: a complex network-based approach. Inf. Sci. 219, 168–180 (2013)

54. Gao, Z., Jin, N.: Complex network from time series based on phase space reconstruction. Chaos Interdiscip. J. Nonlinear Sci. 19(3), 033137 (2009)

55. Zou, Y., Donner, R.V., Marwan, N., Donges, J.F., Kurths, J.: Complex network approaches to nonlinear time series analysis. Phys. Rep. 787, 1–97 (2019)

56. Lacasa, L., Iacovacci, J.: Visibility graphs of random scalar fields and spatial data. Phys. Rev. E 96, 012318 (2017)

57. Lacasa, L., Luque, B., Ballesteros, F., Luque, J., Nuno, J.C.: From time series to complex networks: the visibility graph. Proc. Natl. Acad. Sci. 105(13), 4972–4975 (2008)

58. Gao, Z.-K., Small, M., Kurths, J.: Complex network analysis of time series. EPL (Europhys. Lett.) 116(5), 50001 (2017)

59. Luque, B., Ballesteros, F.J., Robledo, A., Lacasa, L.: Entropy and renormalization in chaotic visibility graphs. Math. Found. Appl. Graph Entropy 6, 1–39 (2016)

60. Luque, B., Lacasa, L., Ballesteros, F., Luque, J., Nuno, J.C.: Horizontal visibility graphs: exact results for random time series. Phys. Rev. E 80(4), 046103 (2009)

61. Nuñez, A.M., Lacasa, L., Gomez, J.P., Luque, B.: Visibility algorithms: a short review. New Front. Graph Theory, 119–152 (2012)

62. Lacasa, L., Luque, B., Luque, J., Nuno, J.C.: The visibility graph: a new method for estimating the Hurst exponent of fractional Brownian motion. EPL (Europhys. Lett.) 86(3), 30001 (2009)

63. Yang, Y., Wang, J., Yang, H., Mang, J.: Visibility graph approach to exchange rate series. Phys. A Stat. Mech. Appl. 388(20), 4431–4437 (2009)

64. Czechowski, Z., Lovato, M., Telesca, L.: Multifractal analysis of visibility graph-based Ito-related connectivity time series. Chaos Interdiscip. J. Nonlinear Sci. 26(2), 023118 (2016)
65. Lacasa, L., Nunez, A., Roldán, É., Parrondo, J.M., Luque, B.: Time series irreversibility: a visibility graph approach. Eur. Phys. J. B 85(6), 1–11 (2012)
66. Rong, L., Shang, P.: Topological entropy and geometric entropy and their application to the horizontal visibility graph for financial time series. Nonlinear Dyn. 92(1), 41–58 (2018)
67. Shang, B., Shang, P.: Directed vector visibility graph from multivariate time series: a new method to measure time series irreversibility. Nonlinear Dyn. 104(2), 1737–1751 (2021)
68. Iacobello, G., Marro, M., Ridolfi, L., Salizzoni, P., Scarsoglio, S.: Experimental investigation of vertical turbulent transport of a passive scalar in a boundary layer: statistics and visibility graph analysis. Phys. Rev. Fluids 4, 104501 (2019)
69. Wu, G., Atilla, I., Tahirin, T., Terziev, M., Wang, L.: Long-voyage route planning method based on multi-scale visibility graph for autonomous ships. Ocean Eng. 219, 108242 (2021)
70. Kong, T., Shao, J., Hu, J., Yang, X., Yang, S., Malekian, R.: EEG-based emotion recognition using an improved horizontal visibility graph. Sensors 21(5), 1870 (2021)
71. Ardalankia, J., Askari, J., Sheykahi, S., Haven, E., Jafari, G.R.: Mapping coupled time-series onto a complex network. EPL (Europhys. Lett.) 132(5), 58002 (2021)
72. Mehraban, S., Shirazi, A., Zamani, M., Jafari, G.: Coupling between time series: a network view. EPL (Europhys. Lett.) 103(5), 50011 (2013)
73. Liu, G., Xiao, F.: A data-driven dynamic data fusion method based on visibility graph and evidence theory. IEEE Access 7, 104443–104452 (2019)
74. Zhang, R., Ashuri, B., Shyr, Y., Deng, Y.: Forecasting construction cost index based on visibility graph: a network approach. Phys. A Stat. Mech. Appl. 493, 239–252 (2018)
75. Lü, L., Zhou, T.: Link prediction in complex networks: a survey. Phys. A Stat. Mech. Appl. 390(6), 1150–1170 (2011)
76. Wen, T., Duan, S., Jiang, W.: Node similarity measuring in complex networks with relative entropy. Commun. Nonlinear Sci. Numer. Simul. 78, 104867 (2019)
77. Liu, W., Lü, L.: Link prediction based on local random walk. EPL (Europhys. Lett.) 89(5), 58007 (2010)
78. Zhang, R., Ashuri, B., Deng, Y.: A novel method for forecasting time series based on fuzzy logic and visibility graph. Adv. Data Anal. Classif. 11(4), 759–783 (2017)
79. Mao, S., Xiao, F.: Time series forecasting based on complex network analysis. IEEE Access 7, 40220–40229 (2019)
80. Mao, S., Xiao, F.: A novel method for forecasting Construction Cost Index based on complex network. Phys. A Stat. Mech. Appl. 527, 121306 (2019)
81. Zhao, J., Mo, H., Deng, Y.: An efficient network method for time series forecasting based on the DC algorithm and visibility relation. IEEE Access 8, 7598–7608 (2020)
82. Liu, G., Xiao, F., Lin, C.-T., Cao, Z.: A fuzzy interval time-series energy and financial forecasting model using network-based multiple time-frequency spaces and the induced-ordered weighted averaging aggregation operation. IEEE Trans. Fuzzy Syst. 28(11), 2677–2690 (2020)
83. Iacovacci, J., Lacasa, L.: Visibility graphs for image processing. IEEE Trans. Pattern Anal. Mach. Intell. 42(4), 974–987 (2019)
84. Pei, L., Li, Z., Liu, J.: Texture classification based on image (natural and horizontal) visibility graph constructing methods. Chaos Interdiscip. J. Nonlinear Sci. 31(1), 013128 (2021)
85. Zhu, D., Semba, S., Yang, H.: Matching intensity for image visibility graphs: a new method to extract image features. IEEE Access 9, 12611–12621 (2021)
86. Zhang, J., Small, M.: Complex network from pseudoperiodic time series: topology versus dynamics. Phys. Rev. Lett. 96(23), 238701 (2006)
87. Xu, X., Zhang, J., Small, M.: Superfamily phenomena and motifs of networks induced from time series. Proc. Natl. Acad. Sci. 105(50), 19601–19605 (2008)
88. Packard, N.H., Crutchfield, J.P., Farmer, J.D., Shaw, R.S.: Geometry from a time series. Phys. Rev. Lett. 45(9), 712 (1980)
89. Gao, J.: Recurrence time statistics for chaotic systems and their applications. Phys. Rev. Lett. 83(16), 3178 (1999)
90. Zhou, T.-T., Jin, N.-D., Gao, Z.-K., Luo, Y.-B.: Limited penetrable visibility graph for establishing complex network from time series. Acta Physica Sinica 61(5) (2012)
91. Lacasa, L., Toral, R.: Description of stochastic and chaotic series using visibility graphs. Phys. Rev. E 82(3), 036120 (2010)
92. Wang, M., Vilela, A.L., Du, R., Zhao, L., Dong, G., Tian, L., Stanley, H.E.: Exact results of the limited penetrable horizontal visibility graph associated to random time series and its application. Sci. Rep. 8(1), 1–13 (2018)
93. Wang, M., Vilela, A.L., Du, R., Zhao, L., Dong, G., Tian, L., Stanley, H.E.: Topological properties of the limited penetrable horizontal visibility graph family. Phys. Rev. E 97(5), 052117 (2018)
94. Ren, W., Jin, N.: Sequential limited penetrable visibility-graph motifs. Nonlinear Dyn. 99(3), 2399–2408 (2020)
95. Gao, Z.-K., Cai, Q., Yang, Y.-X., Dang, W.-D., Zhang, S.-S.: Multiscale limited penetrable horizontal visibility graph for analyzing nonlinear time series. Sci. Rep. 6(1), 1–7 (2016)
96. Lacasa, L., Nicosia, V., Latora, V.: Network structure of multivariate time series. Sci. Rep. 5(1), 1–9 (2015)
97. Nicosia, V., Bianconi, G., Latora, V., Barthelemy, M.: Growing multiplex networks. Phys. Rev. Lett. 111(5), 058701 (2013)
98. Shang, B., Shang, P.: Complexity analysis of multiscale multivariate time series based on entropy plane via vector visibility graph. Nonlinear Dyn. 102(3), 1881–1895 (2020)
99. Kramer, M.A., Eden, U.T., Cash, S.S., Kolaczyk, E.D.: Network inference with confidence from multivariate time series. Phys. Rev. E 79(6), 061916 (2009)
100. Nakamura, T., Tanizawa, T., Small, M.: Constructing networks from a dynamical system perspective for multivariate nonlinear time series. Phys. Rev. E 93(3), 032323 (2016)
101. Gao, Z.-K., Fang, P.-C., Ding, M.-S., Jin, N.-D.: Multivariate weighted complex network analysis for characterizing nonlinear dynamic behavior in two-phase flow. Exp. Therm. Fluid Sci. 60, 157–164 (2015)
102. Gao, Z.-K., Yang, Y.-X., Fang, P.-C., Zou, Y., Xia, C.-Y., Du, M.: Multiscale complex network for analyzing experimental multivariate time series. EPL (Europhys. Lett.) 109(3), 30005 (2015)

103. Ren, W., Jin, N.: Vector visibility graph from multivariate time series: a new method for characterizing nonlinear dynamic behavior in two-phase flow. Nonlinear Dyn. 97(4), 2547–2556 (2019)

104. Lan, X., Mo, H., Chen, S., Liu, Q., Deng, Y.: Fast transformation from time series to visibility graphs. Chaos Interdiscip. J. Nonlinear Sci. 25(8), 083105 (2015)

105. Xu, P., Zhang, R., Deng, Y.: A novel visibility graph transformation of time series into weighted networks. Chaos Solitons Fractals 117, 201–208 (2018)

106. Gao, Z.-K., Jin, N.-D.: A directed weighted complex network for characterizing chaotic dynamics from time series. Nonlinear Anal. Real World Appl. 13(2), 947–952 (2012)

107. Iacovacci, J., Lacasa, L.: Sequential visibility-graph motifs. Phys. Rev. E 93(4), 042309 (2016)

108. Bezsudnov, I., Snarskii, A.: From the time series to the complex networks: the parametric natural visibility graph. Phys. A Stat. Mech. Appl. 414, 53–60 (2014)

109. Gao, Z.-K., Cai, Q., Yang, Y.-X., Dang, W.-D.: Time-dependent limited penetrable visibility graph analysis of nonstationary time series. Phys. A Stat. Mech. Appl. 476, 43–48 (2017)

110. Li, X., Sun, M., Gao, C., Han, D., Wang, M.: The parametric modified limited penetrable visibility graph for constructing complex networks from time series. Phys. A Stat. Mech. Appl. 492, 1097–1106 (2018)

111. Ahmadlou, M., Adeli, H., Adeli, A.: Improved visibility graph fractality with application for the diagnosis of autism spectrum disorder. Phys. A Stat. Mech. Appl. 391(20), 4720–4726 (2012)

112. Marwan, N., Donges, J.F., Zou, Y., Donner, R.V., Kurths, J.: Complex network approach for recurrence analysis of time series. Phys. Lett. A 373(46), 4246–4254 (2009)

113. Donner, R.V., Zou, Y., Donges, J.F., Marwan, N., Kurths, J.: Recurrence networks-a novel paradigm for nonlinear time series analysis. New J. Phys. 12(3), 033025 (2010)

114. Donner, R.V., Small, M., Donges, J.F., Marwan, N., Zou, Y., Xiang, R., Kurths, J.: Recurrence-based time series analysis by means of complex network methods. Int. J. Bifurc. Chaos 21(04), 1019–1046 (2011)

115. Goncalves, B.A., Atman, A.: Visibility graph combined with information theory: an estimator of stock market efficiency. J. Netw. Theory Finance (2017)

116. Dai, P.-F., Xiong, X., Zhou, W.-X.: Visibility graph analysis of economy policy uncertainty indices. Phys. A Stat. Mech. Appl. 531, 121748 (2019)

117. Zhan, T., Xiao, F.: A fast evidential approach for stock forecasting. Int. J. Intell. Syst. (2021). https://doi.org/10.1002/int.22598

118. Zhu, G., Li, Y., Wen, P.: Analysis and classification of sleep stages based on difference visibility graphs from a single-channel EEG signal. IEEE J. Biomed. Health Inform. 18(6), 1813–1821 (2014)

119. Bao, J., Chen, W., Shui, Y.-s., Xiang, Z.-t.: Complexity analysis of traffic time series based on multifractality and complex network. In: 2017 4th International Conference on Transportation Information and Safety (ICTIS), pp. 257–263. IEEE (2017)

120. Zhang, Z., Zhang, A., Sun, C., Xiang, S., Li, S.: Data-driven analysis of the chaotic characteristics of air traffic flow. J. Adv. Transp. 2020 (2020)

121. Kundu, S., Opris, A., Yukutake, Y., Hatano, T.: Extracting correlations in earthquake time series using visibility graph analysis. Front. Phys. 9, 179 (2021)

122. Hloupis, G.: Temporal pattern in corinth rift seismicity revealed by visibility graph analysis. Commun. Nonlinear Sci. Numer. Simul. 51, 13–22 (2017)

123. Hu, J., Xia, C., Li, H., Zhu, P., Xiong, W.: Properties and structural analyses of USA’s regional electricity market: a visibility graph network approach. Appl. Math. Comput. 385, 125434 (2020)

124. Fan, X., Li, X., Yin, J., Tian, L., Liang, J.: Similarity and heterogeneity of price dynamics across China’s regional carbon markets: a visibility graph network approach. Appl. Energy 235, 739–746 (2019)

125. Chen, S., Hu, Y., Mahadevan, S., Deng, Y.: A visibility graph averaging aggregation operator. Phys. A Stat. Mech. Appl. 403, 1–12 (2014)

126. Jiang, W., Wei, B., Zhan, J., Xie, C., Zhou, D.: A visibility graph power averaging aggregation operator: a methodology based on network analysis. Comput. Ind. Eng. 101, 260–268 (2016)

127. Wang, H., Mo, H., Sadiq, R., Hu, Y., Deng, Y.: Ordered visibility graph weighted averaging aggregation operator: a methodology based on network analysis. Comput. Ind. Eng. 88, 181–190 (2015)

128. Xu, P., Zhang, R., Deng, Y.: A novel weight determination method for time series data aggregation. Phys. A Stat. Mech. Appl. 482, 42–55 (2017)

129. Ravetti, M.G., Carpi, L.C., Gonçalves, B.A., Frery, A.C., Rosso, O.A.: Distinguishing noise from chaos: objective versus subjective criteria using horizontal visibility graph. PloS One 9(9), e108004 (2014)

130. Lacasa, L., Flanagan, R.: Time reversibility from visibility graphs of nonstationary processes. Phys. Rev. E 92(2), 022817 (2015)

131. Ahmadlou, M., Adeli, H., Adeli, A.: New diagnostic EEG markers of the Alzheimer’s disease using visibility graph. J. Neural Transm. 117(9), 1099–1109 (2010)

132. Gao, Z.-K., Cai, Q., Yang, Y.-X., Dong, N., Zhang, S.-S.: Visibility graph from adaptive optimal kernel time-frequency representation for classification of epileptiform EEG. Int. J. Neural Syst. 27(04), 1750005 (2017)

133. Supriya, S., Siuly, S., Wang, H., Cao, J., Zhang, Y.: Weighted visibility graph with complex network features in the detection of epilepsy. IEEE Access 4, 6554–6566 (2016)

134. Zhu, G., Li, Y., Wen, P.P., Wang, S.: Analysis of alcoholic EEG signals based on horizontal visibility graph entropy. Brain Inform. 1–4, 19–25 (2014)

135. Telesca, L., Lovallo, M.: Analysis of seismic sequences by using the method of visibility graph. EPL (Europhys. Lett.) 97(5), 50002 (2012)

136. Liu, C., Zhou, W.-X., Yuan, W.-K.: Statistical properties of visibility graph of energy dissipation rates in three-dimensional fully developed turbulence. Phys. A Stat. Mech. Appl. 389(13), 2675–2681 (2010)
137. Ni, X.-H., Jiang, Z.-Q., Zhou, W.-X.: Degree distributions of the visibility graphs mapped from fractional brownian motions and multifractal random walks. Phys. Lett. A 373(42), 3822–3826 (2009)
138. Ahmadlou, M., Adeli, H.: Visibility graph similarity: a new measure of generalized synchronization in coupled dynamic systems. Phys. D Nonlinear Phenomena 241(4), 326–332 (2012)
139. Hayes, M.H.: Statistical Digital Signal Processing and Modeling. Wiley (2009)
140. Farmer, J.D., Sidorowich, J.J.: Predicting chaotic time series. Phys. Rev. Lett. 59(8), 845 (1987)
141. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: beyond efficient transformer for long sequence time-series forecasting. Proc. AAAI Conf. Artif. Intell. 35, 11106–11115 (2021)
142. Lim, B., Zohren, S.: Time-series forecasting with deep learning: a survey. Philos. Trans. R. Soc. A 379(2194), 20200209 (2021)
143. Hu, Y., Xiao, F.: A novel method for forecasting time series based on directed visibility graph and improved random walk. Phys. A Stat. Mech. Appl. 594, 127029 (2022)
144. Liu, F., Deng, Y.: A fast algorithm for network forecasting time series. IEEE Access 7, 102554–102560 (2019)
145. Yager, R.R., Alajlan, N.: Multicriteria decision-making with imprecise importance weights. IEEE Trans. Fuzzy Syst. 22(4), 882–891 (2013)
146. Yager, R.R., Filev, D.P.: Induced ordered weighted averaging operators. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) 29(2), 141–150 (1999)
147. Ashuri, B., Lu, J.: Time series analysis of ENR construction cost index. J. Constr. Eng. Manag. 136(11), 1227–1237 (2010)
148. Latora, V., Nicosia, V., Russo, G.: Complex Networks: Principles, Methods and Applications. Cambridge University Press (2017)
149. Adly, H.M., Moustafa, M.: A hybrid deep learning approach for texture analysis. In: 2017 2nd International Conference on Multimedia and Image Processing (ICMIP), pp. 296–300, IEEE (2017)
150. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Preprint arXiv:1409.1556 (2014)
151. Abdelmounaim, S., Dong-Chen, H.: New Brodatz-based image databases for grayscale color and multiband texture analysis. Int. Scholarly Res. Not. 2013 (2013)
152. Zois, E.N., Zervas, E., Tsourounis, D., Economou, G.: Sequential motif profiles and topological plots for offline signature verification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13248–13258 (2020)
153. Pessa, A.A., Ribeiro, H.V.: Mapping images into ordinal networks. Phys. Rev. E 102(5), 052312 (2020)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.