SCIENTIFIC REPORTS
natureresearch

OPEN

Physically-interpretable classification of biological network dynamics for complex collective motions

Keisuke Fujii[1,2]*, Naoya Takeishi[2], Motokazu Hojo[2], Yuki Inaba[3] & Yoshinobu Kawahara[4,2]

Understanding biological network dynamics is a fundamental issue in various scientific and engineering fields. Network theory is capable of revealing the relationship between elements and their propagation; however, for complex collective motions, the network properties often transiently and complexly change. A fundamental question addressed here pertains to the classification of collective motion network based on physically-interpretable dynamical properties. Here we apply a data-driven spectral analysis called graph dynamic mode decomposition, which obtains the dynamical properties for collective motion classification. Using a ballgame as an example, we classified the strategic collective motions in different global behaviours and discovered that, in addition to the physical properties, the contextual node information was critical for classification. Furthermore, we discovered the label-specific stronger spectra in the relationship among the nearest agents, providing physical and semantic interpretations. Our approach contributes to the understanding of principles of biological complex network dynamics from the perspective of nonlinear dynamical systems.

Complex biological systems are modelled as a collection of the discrete elements that non-linearly interact. Dynamic processes for such complex systems are of great interest in a variety of scientific fields, such as sociology[1,2], epidemiology[3,4], neuroscience[5,6] and physics[7,8]. These systems are often represented as networked systems using graphs, which have several problems such as in classification[9,10], prediction[11,12] and control[13,14]. Among various networked systems, dynamic networks of collective motions[15–17], in which nodes and links are agents and their relationships (e.g. based on their positions), pose several challenges. These challenges arise in changes of the relation among agents (i.e. the structure of a graph) in transient and complex ways, with the exception of moderate changes such as diffusion or contagion[12]. In the network dynamics of collective motions, if the governing equation is given such as mathematical models (e.g. cellular automata[18] and coupled dynamical systems[19]) and controlled systems[20], these are often modeled as graph dynamical systems (GDSs). However, for many biological collective motions[21,22], the relation among agents transiently and complexly changes according to different situations, with the exception of the simply modelled collective motions[23,24]. These biological network dynamics can be regarded as the network dynamics or GDSs of collective motions represented by graph sequence data. To address this problem, conventional approaches have basically computed the properties of a graph in each temporal snapshot[5,25] or in a temporal sliding window[26] of the sequence data. However, for the understanding of network dynamics, these approaches are difficult to directly extract the dynamical properties, i.e. physically-interpretable information about the dynamics such as frequencies with decay/growth rate and the corresponding spatial (network) structures. The fundamental question addressed here is how complex collective motion networks should be classified based on the physically-interpretable dynamical properties.

The motivation of this paper is to understand the principles of network dynamics of biological complex collective motions by directly extracting the dynamical properties of the network in a data-driven manner. As a method of describing nonlinear dynamical systems with a global mode by the direct extraction of dynamical properties, operator-theoretic approaches have attracted attention in fields such as applied mathematics, physics and machine learning. One of the approaches is based on the composition operator (often referred to as the

[1]Graduate School of Informatics, Nagoya University, Nagoya, Japan. [2]RIKEN Center for Advanced Intelligence Project, Tokyo, Japan. [3]Japan Institute of Sports Sciences, Tokyo, Japan. [4]Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan. *email: fujii@i.nagoya-u.ac.jp

Koopman operator[27,28]), which defines the time evolution of observation functions in a function space, rather than directly defines the time evolution in a state space from a classical and popular view of the analysis. The advantage of using the operator-theoretic approach is to lift the analysis of nonlinear dynamical systems to a linear (but infinite-dimensional) regime, which is more amenable to subsequent analysis.

Among several estimation methods, one of the most popular algorithms for spectral analysis of the Koopman operator is dynamic mode decomposition (DMD), which was originally developed in fluid physics[29,30]. DMD has been successfully applied in many real-world problems, such as analyses of epidemiology and electrocorticography[31,32]. Among several variants of DMDs (for details, see Material and Methods), *Graph DMD*[33] can extract and visualise the underlying low-dimensional global dynamics of GDSs with structures among observables from data. However, for more general complex collective motions including strategic human groups, agents flexibly change the rules of behaviour according to the situation; thus, the computation of a feature for the classifier using the obtained spectra cannot be uniquely determined. Therefore, in this paper, we computed Graph DMD for each temporal sliding window and the feature for the classifier reflecting the essential (e.g. dynamical) information for multiple types of collective motions. Previous other approaches have extracted linear dynamics in complex networks[34] and physical interaction networks[35]. Recent neural network approaches automatically extract graph (spatial) and temporal information and perform classification or prediction[36,37]. In contrast to these approaches, our methodological contribution is to provide a classification method by directly extracting physically-interpretable dynamical properties for nonlinear GDSs in a data-driven manner. Additionally, we propose a more straightforward formulation for Graph DMD in an observed data space than the previous one[33].

One of the goals of this data-driven or equation-free method[38] is characterising complex collective motions that emerge from complex rules, rather than motions that emerge from fewer rules such as biological rhythms (e.g. a day and year) and simple rules (e.g. attraction and repulsion). Organised human tasks in small groups such as navigation[21] and ballgame teams[22] provide excellent examples of complex dynamics and pose challenges in many research fields because of their switching and overlapping hierarchical subsystems[22]. In these cases, the focusing networks are often small (e.g. less than 30 nodes) but spatio-temporally complex due to the highly-strategic (or contextual) behaviours. Thus, we hypothesised that the node information (i.e. individual agent) is more critical for the understanding the subtle yet significant differences between motions[22] rather than the network properties such as graph spectrum[39]. The previous works[38,40] have predictively classified the interactions into two group outcomes (i.e. scored or unscored) while reflecting the time-varying interactions among attackers, defenders and the ball. However, since the previous method utilising so-called kernel methods decompose the modes in infinite functional space to acquire high expressiveness[41], it is difficult to physically interpret or visualise the decomposed modes in the observed data space. Thus, the scientific contribution of this study is to provide a method for interpreting the discovered network structures corresponding to the extracted dynamical properties. Additionally, from the viewpoint of practical applications, supervisors such as coaches and researchers who analyse collective motions of agents (e.g. humans, other animals and objects), must exert great effort to observe the motions and label them to specific formations or objectives. The practical advantage of our approach is thus to automatically classify such complex collective motions to aid further analyses such as discovering network structures or dynamical properties.

As already mentioned, the purpose of this study is to understand the network dynamics for complex collective motions via physically-interpretable classification using data-driven spectral analysis via GDSs known as Graph DMD. To this end, we first present the theoretical framework and results of applying graph DMD to actual ballgame data. Then, to validate our approach, we present the classification results of multiple recognition tasks for globally different collective behaviours. Again, we hypothesise that, in addition to physical information, the contextual node information is critical for the classification. We investigated this hypothesis by comparing various existing classification methods. Finally, we visualise and physically and semantically interpret the network structures (called Graph DMD modes) that correspond to extracted dynamics properties.
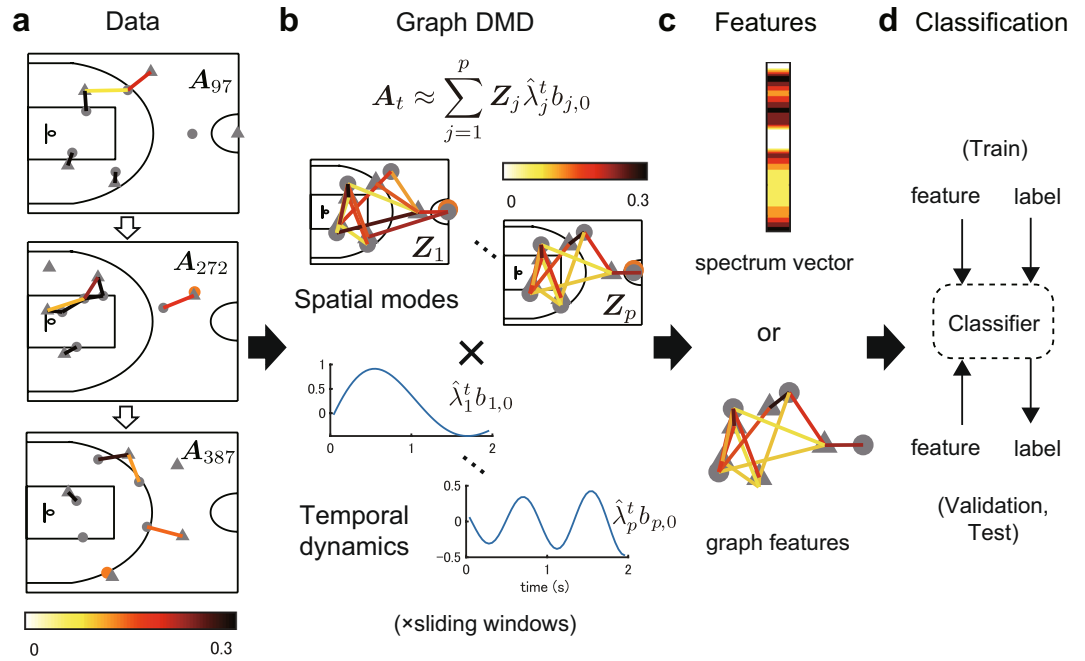
Results

Graph DMD framework. Here, we briefly review Koopman spectral analysis, which is the underlying theory for various DMDs, and then describe the Graph DMD framework. First, we consider a nonlinear dynamical system: $\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t)$, where $\boldsymbol{x}_t$ is the state vector in the state space $\mathcal{M} \subset \mathbb{R}^p$ with time index $t \in \mathbb{T} := \mathbb{N}_0$. *The Koopman operator*, which we denote by $\mathcal{K}$, is a linear operator acting on a scalar observable function $g: \mathcal{M} \to \mathbb{C}$ defined by

$$\mathcal{K} g = g \circ \boldsymbol{f}, \tag{1}$$

where $g \circ \boldsymbol{f}$ denotes the composition of g with $\boldsymbol{f}$[27]. That is, it maps g to the new function $g \circ \boldsymbol{f}$. We assume that $\mathcal{K}$ has only discrete spectra. Then, it generally performs an eigenvalue decomposition: $\mathcal{K} \varphi_j(\boldsymbol{x}) = \lambda_j \varphi_j(\boldsymbol{x})$, where $\lambda_j \in \mathbb{C}$ is the j-th eigenvalue (called *the Koopman eigenvalue*) and φ_j is the corresponding eigenfunction (called *the Koopman eigenfunction*). We denote the concatenation of scalar functions as $\boldsymbol{g} := [g_1, \ldots, g_d]^\top$. If each g_i lies within the space spanned by the eigenfunction φ_j, we can expand the vector-valued $\boldsymbol{g}$ in terms of these eigenfunctions as $\boldsymbol{g}(\boldsymbol{x}) = \sum_{j=1}^\infty \varphi_j(\boldsymbol{x}) \boldsymbol{\psi}_j$, where $\boldsymbol{\psi}_j$ is a set of vector coefficients called *the Koopman modes*. Through the iterative applications of $\mathcal{K}$, the following equation is obtained:

$$\boldsymbol{g}(\boldsymbol{x}_t) = (\boldsymbol{g} \circ \underbrace{\boldsymbol{f} \circ \cdots \circ \boldsymbol{f}}_{t})(\boldsymbol{x}_0) = \sum_{j=1}^\infty \lambda_j^t \varphi_j(\boldsymbol{x}_0) \boldsymbol{\psi}_j. \tag{2}$$

Figure 1. Schematic diagram of Graph DMD and classification. (**a**) Obtained data e.g. in three temporal frames. The colours in each edge represent the values as a function of the distance between agents; they approach 0 if far and 1 if near (further details are provided in the main text). The input of Graph DMD is adjacency matrix series A_t. (**b**) Graph DMD decomposes the input A_t into the sum of the product of graph (spatial) modes and temporal dynamics. To analyse the time-varying dynamics, Graph DMD is performed for each temporal sliding window. (**c**) After Graph DMD, features for classification are computed in different approaches. One is simply to vectorise the Graph DMD modes (i.e. spectrum). The second is to compute graph features using existing methods. Other approaches are described in the main text. (**d**) Using feature vectors and labels, a classification model is trained, validated and tested.

Therefore, λ_j characterises the time evolution of the corresponding Koopman mode ψ_j, i.e. the phase of λ_j determines its frequency and the magnitude determines the growth rate of its dynamics.

Among several possible methods to compute the above modal decomposition from data, DMD[29,30] is the most popular algorithm, which estimates an approximations of the decomposition in Eq. (2). For the details of basic DMD and its variants, see Material and Methods. Among several variants of DMDs, Graph DMD[33] can extract and visualise the underlying low-dimensional global dynamics of GDSs with structures among observables from data. Then, we briefly introduce Graph DMD framework, described in Fig. 1a,b. Here, we consider an autonomous discrete-time weighted and undirected GDS defined as

$$G = (\mathscr{V}, \mathcal{E}, \boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{f}, \boldsymbol{g}_c, \boldsymbol{A}_t), \tag{3}$$

where $\mathscr{V} = \{V^1, \ldots, V^m\}$ and $\mathcal{E} = \{E^1, \ldots, E^l\}$ are the vertex and edge sets of a graph, respectively, fixed at each time $t \in \mathbb{T} := \mathbb{N}_0$. $\boldsymbol{x}_t$ is the state vector in a state space $\mathscr{M} \subset \mathbb{R}^p$ for the GDS and $\boldsymbol{f}: \mathscr{M} \to \mathscr{M}$ is a (typically, nonlinear) state-transition function (again, $\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t)$). $\boldsymbol{y}_t = [\boldsymbol{y}_{1,t}^\top, \ldots, \boldsymbol{y}_{m,t}^\top]^\top$ are concatenated observed values, where $\boldsymbol{y}_{i,t} \in \mathbb{R}^d$ are observed values for a vertex $i = 1, \ldots, m$. They are given by $\boldsymbol{y}_{i,t} := \boldsymbol{g}_c(\boldsymbol{x}_t)$, where $\boldsymbol{g}_c = [\boldsymbol{g}_1^\top, \ldots, \boldsymbol{g}_m^\top]^\top$ is a concatenated vector-valued observable function (i.e. $\boldsymbol{g}_c: \mathscr{M} \to \mathbb{R}^{md}$). $\boldsymbol{A}_t \in \mathbb{R}^{m \times m}$ is an adjacency matrix, whose component $A_{i,j,t}$ represents the weight on the edge between V^i and V^j at each time t. For example, the weight represents the relation (e.g. a function of distance) between moving agents in multi-agent systems.

For the formulation of Graph DMD, we propose a more straightforward formulation with dependent structure among observables than the previous formulation based on vector-valued reproducing kernel Hilbert spaces (RKHSs)[33]. Regarding the details of the analysis, the connection with Graph DMD (such as the relation between $\boldsymbol{A}_t$ and $\boldsymbol{g}_c$) and the reason why it can visualise the relation between elements, see Materials and Methods. For the practical implementation of the spectral decomposition, a modified tensor-based DMD[33,42], which is a generalised DMD for the application to tensor data, is applied to the adjacency matrix series.

In Graph DMD, a sequence of the matrices $\boldsymbol{A}_t$ or an order-3 tensor $\mathscr{A}_{:,:,t}$ for $t = 0, \ldots, \tau$ is given, where colons are used to indicate all elements. In a similar way of basic DMD procedure (see Materials and Methods), we define tensors $\mathscr{X}$, $\mathscr{Y}$ such that $\mathscr{X}_{:,:,t} = \mathscr{A}_{:,:,t}$ and $\mathscr{Y}_{:,:,t} = \mathscr{A}_{:,:,t+1}$ for $t = 0, \ldots, \tau - 1$. Instead of singular value decomposition (SVD) in basic DMD, Graph DMD utilises tensor-train decomposition[43] to maintain the tensor structure of input data. Here, we compute $\boldsymbol{M} \in \mathbb{R}^{m^2 \times r_2}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{r_2 \times r_2}$ and $\boldsymbol{N} \in \mathbb{R}^{r_2 \times \tau}$ by matricising after tensor-train decomposition of $\mathscr{X}$ (r_2 is a tensor-train rank in the decomposition and $\boldsymbol{\Sigma}$ is a full-rank diagonal matrix; for further under-
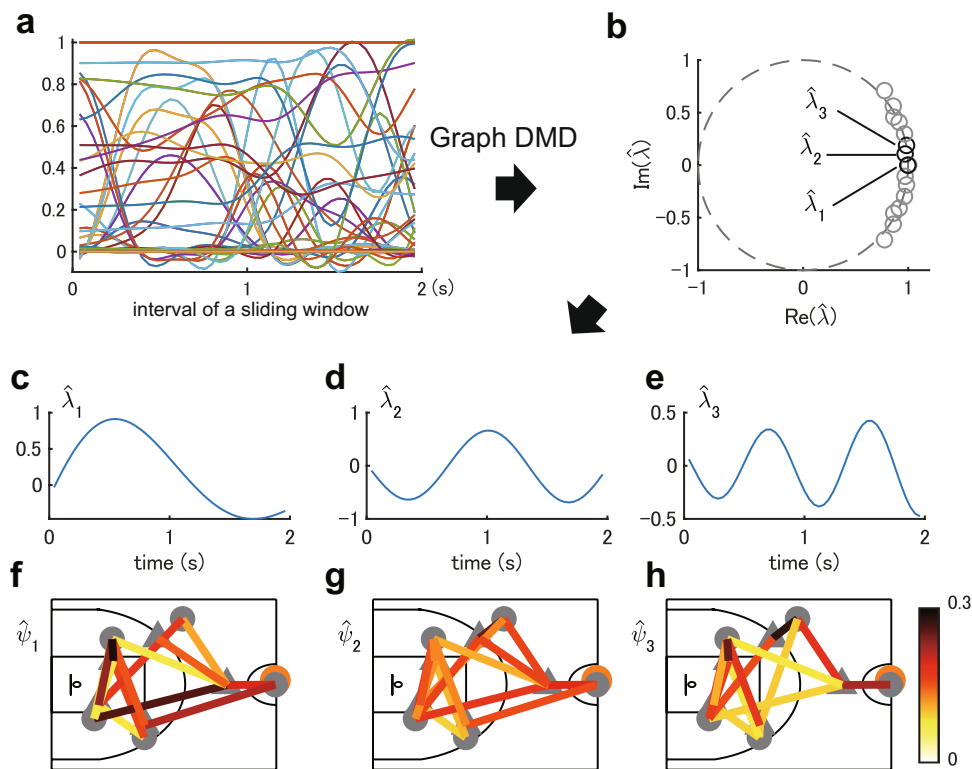
Figure 2. A representative example of Graph DMD. (**a**) An example of obtained data (components of adjacency matrix series). (**b**) Graph DMD eigenvalues (black circles) computed by Graph DMD in a complex plane. (**c–e**) Time dynamics and (**f–h**) visualised Graph DMD modes for the dynamics with the lowest frequencies.

standing, see the previous paper[33]). Similarly, we compute $P \in \mathbb{R}^{m^2 \times s_2}$ and $Q \in \mathbb{R}^{s_2 \times \tau}$ by matricising after tensor-train decomposition of $\mathcal{Y}$, where s_2 is a tensor-train rank. Note that this is similar to SVD in the matrix form, but SVD and this matricisation after tensor-train decomposition with maintaining the tensor structure are completely different. After that, in a similar way of basic DMD procedure, we then define a matrix $\hat{F} = (M^* \cdot P)(Q \cdot N^\dagger)\Sigma^{-1}$, where M^* is the Hermitian transpose of M and $N^\dagger$ is the pseudo-inverse of N. Thereafter, we perform eigendecomposition of $\hat{F}$ and obtain eigenvectors w_j and eigenvalues $\hat{\lambda}_j$ for $j = 1, \ldots, p$. The latter is the estimated Koopman eigenvalues called *Graph DMD eigenvalues*. Finally, we obtain the spatial coefficients $Z_j \in \mathbb{C}^{m \times m}$ by matricising $z_j = (1/\hat{\lambda}_j) \cdot P Q \cdot N^\dagger \Sigma^{-1} \cdot w_j$ in the inputted adjacency matrix form, which are called *Graph DMD modes*. In summary, a sequence of adjacency matrices is decomposed into spatial coefficients and temporal dynamics (Fig. 1b):

$$A_t \approx \sum_{j=1}^{p} Z_j \hat{\lambda}_j^t b_{j,0},$$

(4)

where $b_{j,0}$ works as an initial value described in Materials and Methods (we also describe other detailed procedures there).

Examples of Graph DMD for ballgame data. Next, we show a representative example of Graph DMD results in Fig. 2. Here we used player-tracking data from actual basketball games. Position data was composed of the horizontal Cartesian positions of every player and the ball on the court, recorded at 25 frames per second. We analysed an attack-segment defined as the period that begins when all players enter the attacking side of the court and end before a shot. The input of Graph DMD is adjacency matrix series A_t visualised in Fig. 1a and its elements are:

$$A_{i,j,t} = \exp\left(\frac{-\left\|y_{i,t} - y_{j,t}\right\|_2^2}{2\sigma}\right),$$

(5)

where i and j indicate players and a ring (i.e. a goal as geometric information), $\|\cdot\|_2$ is the Euclidean norm and σ is a coefficient for adjusting the value of $A_{i,j,t}$, which represents a proximity with contextual meaning in this case (for details, see Materials and Methods). Furthermore, since the order of i is not uniquely determined in general, we sorted the adjacency matrix in order of nearest attackers and defenders from the ball in each time stamp, to

	Team-defence			Team-offence		
	Acc	AUC	F-measure	Acc	AUC	F-measure
GDMD spectrum	0.785 ± 0.062	0.833 ± 0.057	0.533 ± 0.058	0.809 ± 0.046	0.803 ± 0.061	0.455 ± 0.060
GDMD Laplacian	0.756 ± 0.045	0.787 ± 0.062	0.506 ± 0.057	0.770 ± 0.047	0.730 ± 0.057	0.405 ± 0.057
GDMD GCN	0.708 ± 0.040	0.678 ± 0.076	0.406 ± 0.078	0.782 ± 0.058	0.626 ± 0.110	0.338 ± 0.069
DMD spectrum	0.765 ± 0.051	0.793 ± 0.068	0.509 ± 0.055	0.794 ± 0.046	0.755 ± 0.062	0.425 ± 0.056
KDMD spetral kernel	0.664 ± 0.052	0.612 ± 0.094	0.407 ± 0.074	0.736 ± 0.067	0.625 ± 0.103	0.350 ± 0.069
Hand-crafted feature	0.756 ± 0.050	0.793 ± 0.040	0.510 ± 0.047	0.745 ± 0.048	0.655 ± 0.076	0.369 ± 0.054
Spatio-temporal GCN	0.711 ± 0.104	0.528 ± 0.086	0.338 ± 0.091	0.782 ± 0.051	0.540 ± 0.077	0.289 ± 0.067

Table 1. Classification performance of seven methods for two recognition tasks. Accuracy (Acc), area under the curve (AUC) based on receiver operating characteristic (ROC) curve and F-measure are indicated. Overall, classification performances in GDMD spectrum were higher than those in most of remaining methods.

provide the order of player to the semantic information. Here, we denote the first to the fifth nearest attackers and defenders to the ball as A1, ..., A5 ($i = 1, \ldots, 5$) and D1, …, D5 ($i = 6, \ldots, 10$), respectively (for the ring, $i = 11$). Also, we denote the relationship between two players, e.g. A1 and D1 as A1-D1.

Firstly, we obtained DMD eigenvalues via Graph DMD. As illustrated in Fig. 2b, all eigenvalues appear to be on the unit circle. Eigenvalue $\hat{\lambda}_j$ is transformed into temporal frequency ω_j such that $\omega_j = \ln(\hat{\lambda}_j)/2\pi\Delta t$, where Δt is a time interval of the discrete time system (i.e. $\Delta t = 1/25$). Then, for each eigenvalue, we obtained the time dynamics in Fig. 2c–e and Graph DMD modes in Fig. 2f–h. For the sake of clarity, only three pairs of the dynamics and modes showing the smallest frequency in this order are presented. In Fig. 2c–e, Graph DMD extracted the dynamics with approximately one, two and three cycles in a sliding window, indicating that two players with these modes approached and left once, twice and three times, respectively. For the Graph DMD modes, we visualised the strength of the spectrum (i.e. coefficient) for each time dynamics. Note that DMD computes complex-valued DMD modes and time dynamics, but here we only present the absolute value and real parts, respectively. In Supplementary Text 2, we describe the selection of parameters for Graph DMD (e.g. tolerance, temporal window size and cutoff frequency) and quantitatively validated the applicability of Graph DMD to the sport data in terms of the reconstruction error.

Classification in various global collective behaviours. Next, we indicate that our methods had better classification performance than most of the existing methods in Table 1 and Fig. 3. As validation, we performed two classification tasks with different global collective behaviours: the team-defence (zone or person-to-person defence) and team-offence (offence with and without screen-play) recognition tasks. By definition, zone and person-to-person defence are exclusive (i.e. players guard their area and opponent players, respectively). However, they are actually mixed based on different situations whereas experienced people can distinguish them. A screen-play is the basic and minimal strategic cooperative play in basketball[44], in which an attacker stands on the course of defence player like a'screen' and prevents the defender from defending another attacker in a legal way. Additional details are provided in Materials and Methods.

After Graph DMD, we performed classification using three different approaches. The first two approaches create feature vectors and the third automatically extracts the features via a neural network approach. The first approach is simply to vectorise the Graph DMD modes (denoted GDMD spectrum) to directly reflect the node (i.e. player) information. For the team-defence recognition task, we used the elements of the modes regarding the relations among defenders, those among attackers and defenders and those among defenders and ring (as geometric information) as a feature vector. For the team-offence recognition task, we used those among attackers and among attackers and defenders (without geometric information). Selection and elimination were based on prior knowledge of the sport. The final two approaches computed graph features using existing methods: the second is graph Laplacian eigenvalues[39] as a baseline feature of a graph[45] (denoted GDMD Laplacian) and the third is deep graph convolutional neural networks[46] (denoted GDMD GCN) as a recent promising method for classifying the graph data. The second approach extracts the graph topology without node information, and the third one retains more node information and learns the graph topology.

Due to the highly-strategic (or contextual) behaviours in this experiment, we hypothesised that, in addition to physical information, the contextual node information (i.e. the nodes themselves in the first approach) is more important for the classification than the graph features (i.e. the second and third approaches). For all classification methods with the exception of the neural network approaches, we adopted logistic regression as a simple linear binary classification model.

As comparable methods for classifying this data, we adopted four methods. The first is a method using vectorised basic DMD modes[47] (denoted DMD spectrum) as a baseline of DMD approaches. The second is the Koopman spectral kernel[40] using DMD with reproducing kernels[41] as the existing method for classifying the collective motion dynamics[38] (denoted KDMD spectral kernel). For these two methods, the input data must be a matrix; thus we did not utilise the structure of the graph sequence data. The third is a hand-crafted feature as a simple baseline method, consisting of the vectorised temporal average, maximum and minimum values of the elements of the input adjacency matrix series. Fourth is an advanced neural network approach for classifying graph sequence data called spatio-temporal GCN[36] as a recent promising end-to-end method. More information on the selection and the details of these methods are provided in Supplementary Text 1.
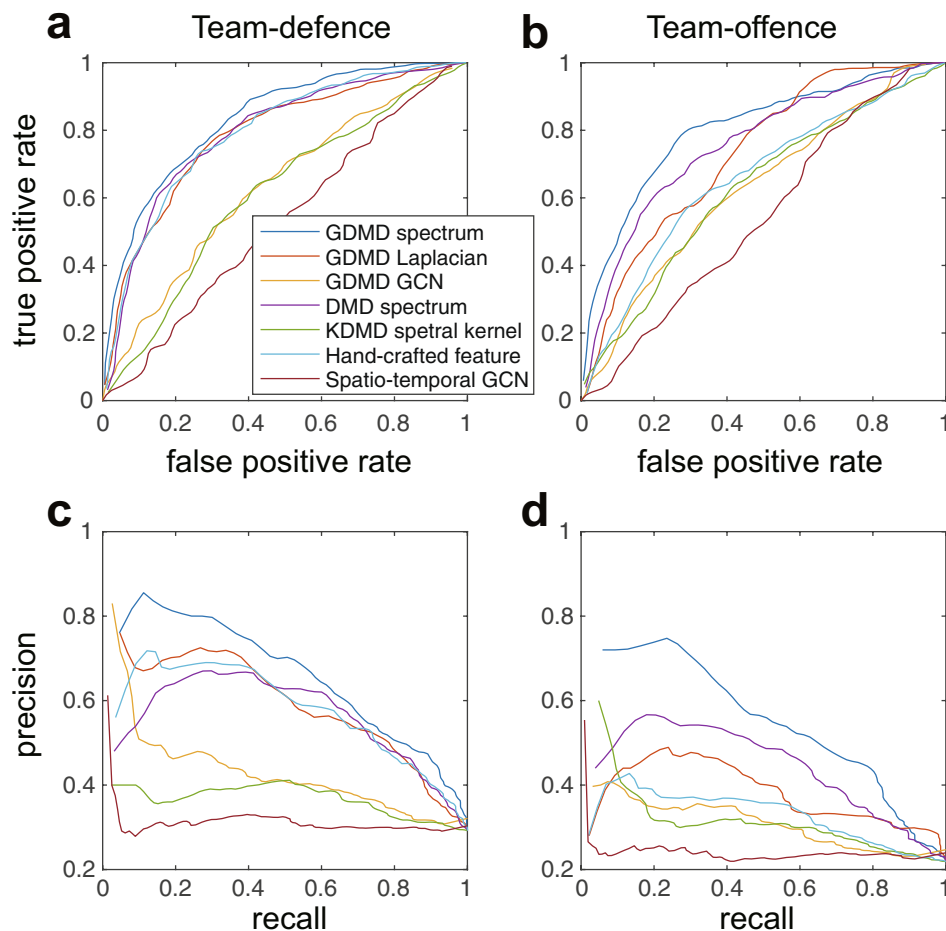
Figure 3. Classification results of seven methods in two recognition tasks. Receiver operating characteristic (ROC) curves in (**a**) the team-defence and (**b**) team-offence recognition task are shown. Also, trade-off curves between recall and precision in (**c**) the team-defence and (**d**) team-offence recognition task are shown. The seven methods are given in the legend of (**a**) and the text.

We investigated classification performance in Table 1 in terms of accuracy, the area under the curve (AUC) based on receiver operating characteristic (ROC) curve in Fig. 3a,b, and F-measure, which is the trade-off between recall and precision (the curve is shown in Fig. 3c,d). Overall, the classification performance in GDMD spectrum was better than those in most of the remaining methods. In the statistical evaluation, there were significant differences in all classification performance and tasks ($p > 3.63 \times 10^{-8}$) using analysis of variance (ANOVA) or Friedman test. In the following evaluations, we indicate the post-hoc comparison results. Accuracy in GDMD spectrum was significantly higher than that of other methods ($r > 0.39$ and $p < 0.03$) for both recognition tasks, with the exception of GDMD Laplacian and DMD spectrum in the team-defence recognition ($r > 0.16$ and $p > 0.05$). AUC and F-measure in GDMD spectrum were also higher than those of the other methods ($r > 0.40$ and $p < 0.028$) for both recognition tasks, with the exception of hand-crafted feature in the team-defence recognition (both AUC and F-measure: $r > 0.34$ and $p > 0.05$). As a feature extraction method, Graph DMD was a better method than most of the other methods, and especially simple Graph DMD spectrum was better than the graph features in the collective motions.

Analysis of visualised Graph DMD modes. Since our method can decompose the data into temporal dynamics and spatial coherent structures in the observed data space, we can interpret the decomposed modes that contribute to the classification results. Here we show the averaged spectra for each label and classification task to identify the trends (note that the location of collective motion networks shown in Fig. 2 change with time; thus for an accurate understanding, we averaged and visualised them in an adjacency matrix form).

For the team-defence recognition task, a strong spectrum for zone defence in Fig. 4a was observed in the relationship among the nearest defenders (expanded in the figure) but it was not observed in person-to-person defence Fig. 4b. The variability of the feature vectors is also presented via boxplots in Fig. 5a–c. In examining the individual contribution, statistical analysis using logistic regression indicates that the relationships among defenders (D1-D2 and D4-D5: odds ratio > 31.81 and $p < 0.023$; D1-D3 was also significant but too small), those among attackers and defenders (A1-D4, A1-D5, A2-D5, A4-D1, and A5-D5: odds ratio > 0.02 and $p < 0.023$)
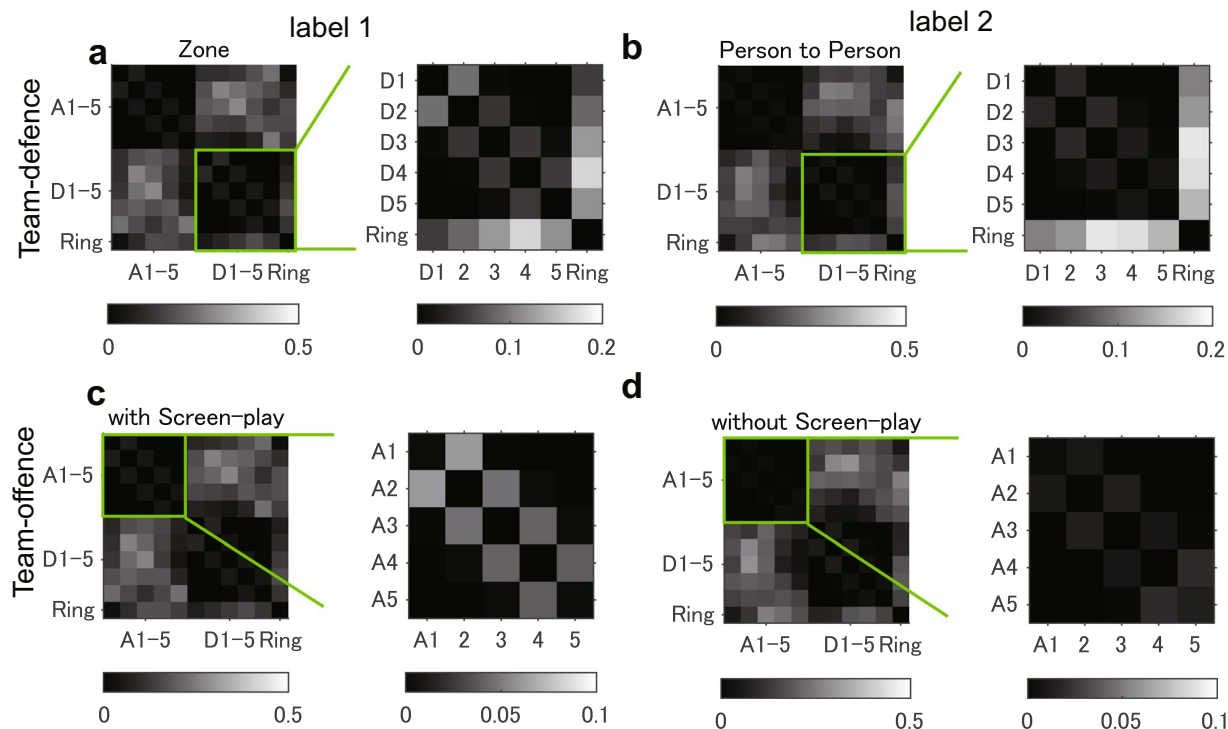
Figure 4. Averaged Graph DMD modes. Averaged Graph DMD modes for label 1 (**a** and **c**) and 2 (**b** and **d**) in the team-defence (**a** and **b**) and offence (**c** and **d**) recognition tasks are shown. We denote the first to fifth nearest attacker and defender as A1, …, A5 and D1, …, D5, respectively. R indicates the ring (goal). A strong spectrum for (**a**) zone defence was observed in the relationship among the nearest defenders; however, it was not observed in (**b**) person-to-person defence. Similarly, a strong spectrum for (**c**) offence with screen-play was observed in the relationship between the nearest attackers; however, it was not observed for (**d**) offence without screen-play.
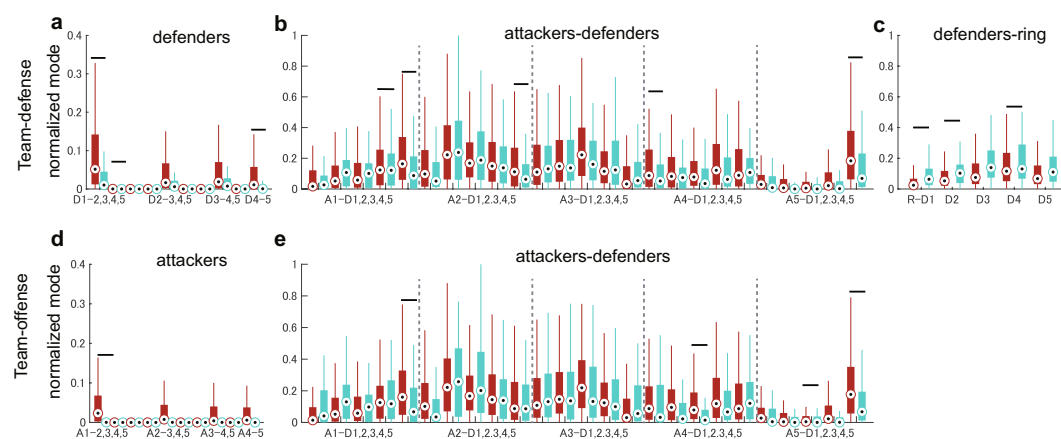
Figure 5. Boxplots of feature vectors as elements of Graph DMD modes. Feature vectors as elements of Graph DMD modes for label 1 (red) and 2 (green) regarding the feature vectors used in the team-defence (**a–c**) and offence (**d** and **e**) recognition tasks are shown (label 1: zone defence and offence with screen-play; label 2: person-to-person defence and offence without screen-play). For the team-defence recognition task, the elements of GDMD modes (**a**) among defenders, (**b**) those among attackers and defenders and (**c**) those among defenders and the ring were selected. For the team-offence recognition task, those (**d**) among attackers and (**e**) those among attackers and defenders were selected. Notations are same as Fig. 4. Horizontal bars indicate that the relationship is significantly explained by logistic regression. For all statistical results, see Supplementary Tables 1 and 2.

and those among defenders and ring (ring-D1, D2 and D4: odds ratio > 0.002 and $p < 0.049$) significantly explained the labels of team-defences. Note that this analysis separately investigated the individual contribution as well as the above classifications. For more detailed statistical results and boxplots of other elements of DMD modes, see Supplementary Table 1 and Supplementary Fig. 2, respectively.

For the team-offence recognition task, a strong spectrum for the offence with screen-play in Fig. 4c was observed in the relationship between the nearest attackers; however, it was not observed for the offence without screen-play Fig. 4d (the boxplot is shown in Fig. 5d,e). Results of logistic regression indicate that the relationship in D1-D2 (odds ratio $= 1.06 \times 10^4$ and $p = 5.36 \times 10^{-5}$) and those among attackers and defenders (A1-D5, A4-D3 and A5-D5: odds ratio > 23.59 and $p < 0.009$; A5-D3 was also significant but too small) significantly explained the labels of team-offence. For more details, see Supplementary Table 2 and Supplementary Fig. 2. With these methods, we can interpret the types of interaction between agents (and environments) that contributes to focusing dynamic properties (i.e. frequency in this case) with a physical meaning. In Discussion section, we discuss further semantic interpretation.

Discussion

The objective of this paper was to understand the network dynamics for complex collective motions via physically-interpretable classification using data-driven spectral analysis of GDSs called Graph DMD. In this section, we discuss the comparison with the conventional approaches, semantically interpret the visualised Graph DMD modes, describe the limitation and the future perspectives for our methods. We then present our conclusions.

Using ballgame data as an example, we classified the strategic collective motions in the team-defence and offence recognition tasks more successfully than most of the existing methods. The results of the comparison with graph features (GDMD Laplacian and GCN) suggest that the collective motions in this case can be classified with the node information (nearest to the ball) rather than the graph features. The results of other DMD methods (DMD spectrum and KDMD spectral kernel) suggest that both the reflecting graph structure and the use of appropriate sliding windows were important for recognition. Furthermore, in comparison with KDMD spectral kernel[40] (i.e. decomposition in a feature space), the proposed method has an advantage in physically and semantically interpreting the DMD modes in the observed data space. Regarding an end-to-end neural network approach (i.e. spatio-temporal GCN), the model for motion capture data (i.e. joint location data for a few people) must be customised. However, each agent is not physically connected in this case; thus, the positional relationship (i.e. constraint condition) dynamically changes in contrast to the motion capture data. In this paper, we used a simple setting with minimal correction; however, further adaptation to complex collective motions is required (e.g. more appropriate customisation of the network and adequate learning such as using a larger amount of data). For hand-crafted features, we compute simple features to demonstrate the validity of our methods in this paper. The use of a more customised framework[44] which includes the detection of a few related people might improve specific classification performance. However, in this paper, we proposed a more generalised classification framework for global dynamics of collective motions, which successfully performed two different recognition tasks (team-defence and offence) by selecting only the elements of the Graph DMD modes as feature vectors. Furthermore, our method can obtain the dynamical properties of the collective motion network.

The proposed method can obtain physically-interpretable dynamical properties of network dynamics and can perform classification even for complex collective motions. For example, in Fig. 5, we found the label-specific stronger spectra in two recognition tasks, in which the relationships among the nearest attackers or defenders, those among attackers and defenders far from the ball and those between defenders and the ring, which were oscillated in low-frequency modes. Semantically, in zone defence, defenders adjusted their positions more according to a teammate (Fig. 5a), the attacker with the ball (Fig. 5b: especially in the distant case from the ball) and less according to the geometric reference point (Fig. 5c) than the defenders in person-to-person defence. For team-offence with screen-play, attackers adjusted their positions more according to a teammate (Fig. 5d) and the distant defender from the ball (Fig. 5e) than the attackers in team-offence without screen-play. Generally, our approach can be applied to the analysis of complex global dynamics in groups of living organisms or artificial agents, which currently eludes mathematical formulation. The human group in a sport used in this study can be considered as an example in which communication is likely to be measured in a physical space[38]. Therefore, by measuring the outputs of communication and assuming that there are underlying dynamics behind the obtained data, our approach can handle various means of communication between agents. In practice, in various sports teams or other human communities, supervisors (e.g. experimenters, coaches and teachers) spend considerable amounts of time analysing the collective motions in their domain. Application of a system, such as the one presented here, can lead to the creation of useful plans that are currently derived only from their implicit experience.

However, there are some limitations to this study. One is the validation of whether our approach can extract true dynamical properties if used as an equation-free method, which cannot confirm the true dynamical properties (e.g. frequencies with growth/decay rate), as well as in the previous works[31,32]. Originally, DMD demonstrates its strength for the dynamical systems which can be mathematically defined[41,48] or of which solutions are empirically known[29,30]. We instead validated our approach using classification performance, a qualitative evaluation with specific knowledge of the sport domain (e.g. low-frequency band) and the reconstruction error (see Materials and Methods). However, a general quantitative validation method for the unknown dynamics is further required. Another is to reflect the more local interaction dynamics such as local competitive and cooperative play by the attackers and defenders[22,49–51], which can provide more practical information in the sport domain. Although the purpose of Graph DMD is to extract the underlying global dynamics of GDSs and we can obtain the interpretable local spectra in the DMD modes, there are other approaches for extracting the more specific local dynamics, such as using heuristic selection of Graph DMD modes and additional hand-crafted features (e.g. specific geometrical information). Even when using only players' location data such as in this study, such methods reflecting local competition can be applied to more practical application such as score prediction[38,40] and prediction of a player to obtain the ball after shot[52].

In conclusion, we applied a data-driven spectral analysis called Graph DMD, which obtains physically-interpretable dynamical properties of network dynamics for the classification even for complex

collective motions. We classified the motions in different global behaviours and discovered that, in addition to the physical properties, the contextual node information was critical for classification. Furthermore, we discovered the label-specific stronger spectra in the relationship between the nearest agents, which provided physical and semantic interpretation. Potential applications to other collective motions are related to the motions which are difficult to discretize strategic motions such as football and handball, or other collective motions such as pedestrians[24] or animal group behaviour[38,53]. Our approach contributes to the understanding of complex networks involving collective motions from the perspective of nonlinear GDSs.

Materials and Methods

Positional data in a ballgame.   The positional data of players and the ball (25 frames per second) were obtained from actual men's Asian international level practical games held in 2015 and preprocessed by STATS SportVU system (Northbrook, IL, USA). We obtained the consent to use it for research. We analysed 220 min of play (in 4 days) in which the two teams scored 746 points (386 vs. 360). For each day, players performed one and a half games (i.e. 60 min) except for 1 day (only one game). The positional data contained the XY position of each player and the XYZ coordinates of the ball on the court (All-court: 28×15 m; half-court: 14×15 m). After the following data segmentation, we obtained 319 attack-segments.

Data segmentation.   Prior to data segmentation, we used Butterworth low-pass filter at 2 Hz and a custom-made automatic individual play-detection system to detect shots using positional data similar to that used in previous studies[22,38,44]. We analysed an attack-segment defined as the period that begins when all players enter the attacking side of the court and end before a shot (we analysed only the attack-segment finishing with a shot). Then, the authors, who have experience in playing and coaching basketball, manually labelled all attack-segments into a zone defence, a person-to-person defence as the team-defence recognition task and offence with and without screen-plays as the team-offence recognition task.

Person-to-person defence is a team-defence strategy in which defenders basically guard their predetermined attackers. In contrast, zone defence is another team-defence strategy in which defenders basically guard their predetermined areas. Although zone and person-to-person defence are exclusive by definition, they are actually mixed according to different situations. For example, against an attacker's penetration with the ball to the ring, the defenders in person-to-person defence also guard the area near the ring; in contrast, against an attacker's shot, the defenders in zone defence also guard the attacker with the ball. Despite this ambiguity, people with the experience of games can distinguish the two team-defence strategies by observing the defensive motion without the ball.

Screen-play is the basic and minimal strategic cooperative play in basketball[44], in which an attacker stands on the course of defence player like a 'screen' and prevents the defender from defending another attacker in a legal way. We labelled attack-segments into an offence with (at least including a screen-play) and without screen-play. Note that we focused on the global network dynamics of collective motions, thus we did not segment and label screen-plays themselves (i.e. in a spatio-temporally accurate sense such as in the previous study[44]).

We randomly created a validation dataset (30 attack-segments) for the selection of parameters regarding Graph DMD, and a test dataset (289 attack-segments) for the subsequent analyses.

Creating adjacency matrix series.   Next, we compute the adjacency matrix sequence in the attack-segment for inputting to the subsequent graph DMD. Here, using the Gaussian kernel in Eq. (5), we converted the distances between all individuals and the geometric reference position (i.e. the ring) into weights of adjacency matrix series, which indicate 1 if close to each other and 0 if far from each other. In this way, we can obtain more stable time series than those directly using the distances. The numerator in the exponential in Eq. (5) represents the distance between all individuals and the ring. Since the order of i cannot be uniquely determined in general, we must define the order based on reasonable grounds. For example, the order of playing positions (e.g. guard, forward and centre) or jersey numbers does not provide a fair comparison among attack-segments because the players themselves frequently changes throughout attack-segments. A simple way to determine the order is based on the distance between players and a reference point (e.g. ball or ring). However, if the order is simply determined by proximity to the reference point, there may arise a problem that, for example, the distance between the third and fifth players is shorter than that between those (the third and fifth) and the fourth players. Thus, first, we simply set $i = 1, 2$ for the two attackers in order from the attacker closer to the ball. Then, we set $i = 3$ for the closest attacker in terms of the sum of the distance to the above two attackers. In this way, we set $i = 4, 5$ for the closest attacker based on the sum of the distance to the three and four attackers, respectively. Similarly, we set $i = 6, ..., 10$ for the five defenders and set $i = 11$ for the ring position to take the geometric information into consideration.

Next, denominator 2σ in the exponential in Eq. (5) is a coefficient for adjusting the value of $A_{i,j,t}$. In this paper, we set σ for the relationship between players that satisfies $A_{i,j,t} = 0.5$ when $\|y_{i,t} - y_{j,t}\|_2 = 1.5$ m (i.e. $\sigma = 1.5^2/2\log 2$), which is considered to be a meaningful proximity between players (e.g. 1 m is near and 2 m is far in terms of the contact of two players). For the additional analysis regarding σ and the ring information, see Supplementary Text 3. Also, we set σ for the relation between players and the ring that satisfies $A_{i,j,t} = 0.5$ when $\|y_{i,t} - y_{11,t}\|_2 = 6$ m (i.e. $\sigma = 6^2/2\log 2$). In this way, a sequence of the matrix A_t with $A_{i,j,t}$ as an element at time t is created as a sequence of adjacent matrices of the graph. In this paper, we used this adjacency matrix series for both team-defence and offence recognition tasks.

Selection of an appropriate representation of the data is a fundamental problem in pattern recognition. Time-series data is challenging to design features for because of difficulties in reflecting the data structure (including time length). A simple method involves using the representative values (e.g. average and maximum values).

However, for time series data, which cannot be distinguished by these representative values, it is necessary to extract the dynamic properties of the time series. One of the effective methods is DMD[29,30] as described below.

Basic DMD and its variants.   Here, we first describe the basic DMD procedure and briefly explain the variants of DMDs including Graph DMD. Among several possible methods to compute the spectral decomposition from data, DMD[29,30] is the most popular algorithm, which estimates an approximations of the decomposition in Eq. (2). Consider a finite-length observation sequence $\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_\tau$ $(\in \mathbb{C}^m)$, where $\boldsymbol{y} := \boldsymbol{g}(\boldsymbol{x}_t)$. Let $\boldsymbol{X} = [\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{\tau-1}]$ and $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_\tau]$. Then, DMD basically approximates it by calculating the eigendecomposition of matrix $\boldsymbol{F} = \boldsymbol{Y}\boldsymbol{X}^\dagger$, where $\boldsymbol{X}^\dagger$ is the pseudo-inverse of $\boldsymbol{X}$. The matrix $\boldsymbol{F}$ may be intractable to analyse directly when the dimension is large. Therefore, in the popular implementation of DMD called exact DMD[47], a rank-reduced representation $\hat{\boldsymbol{F}}$ based on SVD is applied. That is, $\boldsymbol{X} \approx \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$ and $\hat{\boldsymbol{F}} = \boldsymbol{U}^*\boldsymbol{F}\boldsymbol{U} = \boldsymbol{U}^*\boldsymbol{Y}\boldsymbol{V}\boldsymbol{\Sigma}^{(-1)}$, where $*$ is the conjugate transpose. Thereafter, we perform eigendecomposition of $\hat{\boldsymbol{F}} \in \mathbb{C}^{p \times p}$ to obtain the set of the eigenvalues $\hat{\lambda}_j$ and eigenvectors $\boldsymbol{w}_j$. Then, we estimate the Koopman modes in Eq. (2): $\boldsymbol{\psi}_j = \hat{\lambda}_j^{(-1)}\boldsymbol{Y}\boldsymbol{V}\boldsymbol{\Sigma}^{(-1)}\boldsymbol{w}_j \in \mathbb{C}^{m \times p}$, which is called *DMD modes*. Time dynamics of jth mode is defined as $\hat{\lambda}_j^t b_{j,0}$, where $b_{j,0} = \boldsymbol{\psi}_j^\dagger \boldsymbol{y}_0$ and $\boldsymbol{\psi}_j^\dagger$ is the j-th row of the pseudo-inverse of $[\boldsymbol{\psi}_1 \cdots \boldsymbol{\psi}_p]$[54]. In summary, obtained time series data is decomposed into spatial coefficients and temporal dynamics: $\boldsymbol{y}_t \approx \sum_{j=1}^p \boldsymbol{\psi}_j \hat{\lambda}_j^t b_{j,0}$.

Theoretically, for DMD to compute the decomposition Eq. (2), each observable function g_i should lie within the space spanned by the Koopman eigenfunction φ_j, i.e. the data should be rich enough to approximate the eigenfunctions. However, in basic DMD algorithms naively using the obtained data such as exact DMD, the above assumption is not satisfied such as when the data dimension is too small to approximate the eigenfunctions. Thus, there are several algorithmic variants of DMDs to overcome the problem of the original DMD such as a formulation in RKHSs[41], in a multitask framework[53], and using a neural network[48,55]. The previous Graph DMD[33] utilises the structure of graph sequences (i.e. a tensor structure of adjacency matrix series) by a formulation in a vector-valued RKHS and an implementation using tensor-train decomposition. Although the previous Graph DMD is based on a general formulation in a vector-valued RKHS, in this paper, we propose a more straightforward formulation of Graph DMD in an observed data space.

Koopman spectral analysis with dependent structure among observables for Graph DMD.
Here, we propose a more straightforward formulation for Graph DMD called *Koopman spectral analysis with dependent structure among observables* than the previous one[33]. Notations are the same as those in Results. The previous formulation of Graph DMD[33] was based on vector-valued RKHSs endowed with a symmetric positive semi-definite kernel matrix[56]. Although the spectral decomposition of the vector-valued feature function in a vector-valued RKHS is generally formulated, in this study, we propose a more straightforward formulation connecting Koopman spectral analysis and Graph DMD (in Eq. (4)) in an observed data space (i.e. in an adjacency matrix form). First, again, consider a vector-valued observable function $\boldsymbol{g}_c = [\boldsymbol{g}_1^\top, \ldots, \boldsymbol{g}_m^\top]^\top$. To analyse a GDS with dependent structures among the elements of the observable function, $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_m$, we newly consider a matrix-valued observable function $\boldsymbol{G}: \mathscr{M} \to \mathbb{R}^{m \times m}$ as

$$[\boldsymbol{G}(\boldsymbol{x})]_{i,j} = h(\boldsymbol{g}_i(\boldsymbol{x}), \boldsymbol{g}_j(\boldsymbol{x})), \tag{6}$$

for $i, j = 1, \ldots, m$, where h is a function defined in the right-hand side of Eq. (5). Next, we consider a vector-valued function $\boldsymbol{g}_G: \mathscr{M} \to \mathbb{R}^{m^2}$ defined as $\boldsymbol{g}_G(\boldsymbol{x}) = \text{vec}(\boldsymbol{G}(\boldsymbol{x}))$. In analogous ways of a basic Koopman spectral analysis (described in Results), the Koopman operator $\mathscr{K}_G$ is defined by $\mathscr{K}_G \boldsymbol{g}_G = \boldsymbol{g}_G \circ \boldsymbol{f}$, where $\boldsymbol{g}_G \circ \boldsymbol{f}$ denotes the composition of $\boldsymbol{g}_G$ with $\boldsymbol{f}$. We assume that $\mathscr{K}_G$ has only discrete spectra. Then, it generally performs an eigenvalue decomposition: $\mathscr{K}_G \varphi_j(\boldsymbol{x}) = \lambda_j \varphi_j(\boldsymbol{x})$, where $\lambda_j \in \mathbb{C}$ is the j-th eigenvalue and φ_j is the corresponding eigenfunction. Note that the underlying nonlinear dynamical system is the same as the basic formulation, i.e. $\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t)$ and $\boldsymbol{x}_t \in \mathscr{M} \subset \mathbb{R}^p$. Then, if each element $[\boldsymbol{g}_G]_i$ for $i, \ldots, m^2$ lies within the space spanned by the eigenfunction φ_j, we can expand the vector-valued $\boldsymbol{g}_G$ in terms of these eigenfunctions as $\boldsymbol{g}_G(\boldsymbol{x}) = \sum_{j=1}^\infty \varphi_j(\boldsymbol{x})\boldsymbol{\psi}_j$, where $\boldsymbol{\psi}_j$ is a set of vector coefficients. Through the iterative applications of $\mathscr{K}_G$, the following equation is obtained:

$$\boldsymbol{g}_G(\boldsymbol{x}_t) = (\boldsymbol{g}_G \circ \underbrace{\boldsymbol{f} \circ \cdots \circ \boldsymbol{f}}_{t})(\boldsymbol{x}_0) = \sum_{j=1}^\infty \lambda_j^t \varphi_j(\boldsymbol{x}_0)\boldsymbol{\psi}_j. \tag{7}$$

For the practical implementation of the spectral decomposition of the linear operator, it is needed to project data onto directions that are effective in capturing the properties of data such as in exact DMD[47], DMD with reproducing kernels[41] and tensor-based DMDs[33,42]. Here, we compute the projection onto some orthogonal directions based on tensor-train decomposition[43], in the same ways of modified tensor-based DMD[33]. In other words, Graph DMD assumes that the data are sufficiently rich and thus a set of the orthogonal direction gives a good approximation of the representation with the eigenfunctions of $\mathscr{K}_G$. Concretely, for an implementation of the above analysis, we first regard the given or calculated matrices as a realisation of the structure of matrix-valued observable $\boldsymbol{G}(\boldsymbol{x}_t)$ at each t. We denote the realised matrices (i.e. adjacency matrix) as $\boldsymbol{A}_t \in \mathbb{R}^{m \times m}$ for $t = 0, \ldots, \tau$. Second, we need to compute the projection onto orthogonal directions $\boldsymbol{M}$ (see below and Results) and obtain DMD solution $\hat{\boldsymbol{F}} \in \mathbb{R}^{p \times p}$. Then, after eigendecomposition of $\hat{\boldsymbol{F}}$, we compute DMD eigenvalues $\hat{\lambda}_j$ and DMD modes $\boldsymbol{z}_j \in \mathbb{C}^{m^2}$ for $j = 1, \ldots, p$. Finally, we obtain Graph DMD modes $\boldsymbol{Z}_j \in \mathbb{C}^{m \times m}$ by matricising $\boldsymbol{z}_j$.

It should be noted that, although we vectorise $\boldsymbol{G}(\boldsymbol{x}_t)$ in this formulation, we uniquely determine the order of the vector; thus it reflects the dependent structure of the matrix-valued observable, $\boldsymbol{G}(\boldsymbol{x}_t)$. In the implementation,

we utilise tensor-train decomposition to obtain the matrix $\boldsymbol{M} = \boldsymbol{X}\boldsymbol{N}^{\dagger}\boldsymbol{\Sigma}^{-1}$, where $\boldsymbol{X} = [\text{vec}(\boldsymbol{A}_0), \ldots, \text{vec}(\boldsymbol{A}_{\tau-1})]$ (see Results). Although $\boldsymbol{A}_t$ is also vectorised, unlike SVD, $\boldsymbol{M}$ reflects order-3 tensor structure based on tensor-train decomposition. Therefore, both Koopman spectral analysis with dependent structure among observables and Graph DMD can be calculated without breaking the dependent structure.

Computation of Graph DMD.   In this subsection, we describe the implementation of Graph DMD[33]. The procedure to compute Graph DMD modes is described in Results. The Matlab code we used is available at https://github.com/keisuke198619/GraphDMD. To compute this, a modified tensor-based DMD[33,42] using tensor-train decomposition[43] is applied. The tensor-train decomposition is considered to be relatively stable and scalable for high-order tensors compared with the other tensor decomposition methods[43]. The basic algorithm of tensor-train decomposition for an order-N tensor (i.e. $\mathscr{A} \in \mathbb{C}^{n_1 \times \cdots \times n_N}$, where n_l denotes the dimensionality of the l-th mode for $l = 1, \ldots, N$) decompose into N core tensors $\mathscr{A}^{(l)} \in \mathbb{C}^{r_{l-1} \times n_l \times r_l}$, where $r_0 = r_N = 1$, by serial matricisations and SVDs. The input tensor of Graph DMD is an order-3 tensor; thus it is decomposed into a matrix, an order-3 tensor, and a matrix.

For obtaining the time dynamics shown in Eq. (4) and Figs. 1 and 2, we need to compute $b_{j,0}$. Since here we do not need to consider the tensor structure in the time dynamics (i.e. they are univariate time-series), we compute $b_{j,0}$ in a similar way of exact DMD by matricising the DMD mode $\mathscr{Z}_{:,:,j} = \boldsymbol{Z}_j$ (such that the row is the time stamp) and vectorising the initial value $\boldsymbol{A}_0$. After computing the time dynamics, we can compute the reconstructed data shown in Eq. (4). Using the reconstructed data, we eliminated invalid sliding windows (described below) and investigated the validity of the decomposition. In this paper, as a criterion of valid decomposition, we used variability accounted for (VAF) which is commonly used in non-orthogonal dimensionality reduction[57]. VAF is defined as the square error of the reconstructed data and the original data such that $\text{VAF}_j = 1 - (\|\boldsymbol{A} - \hat{\boldsymbol{A}}_j\|_F^2)/\|\boldsymbol{A}\|_F^2$, where $\boldsymbol{A}$ and $\hat{\boldsymbol{A}}$ are matricised input data tensor $\mathscr{A}$ (such that the row is time) and its reconstructed tensor for the $j$the mode, and $\|\cdot\|_F$ is the Frobenius norm. In this paper, if the maximum $\text{VAF}_j < 0.01$, the sliding window was eliminated from the subsequent analyses. We confirmed all attack-segments have at least one valid sliding window.

Next, we investigated the validity of the decomposition when changing the parameters (e.g. the Graph DMD tolerance, sliding window size and cutoff frequency) using a validation dataset. Here we describe the summary (for the details, see Supplementary Text 2). First, we set Graph DMD tolerance ε (i.e. the tolerance in the successive SVD in tensor-train decomposition) to 1.0×10^{-5} based on the result in Supplementary Fig. 1a. For the sliding temporal windows, we set the window size to 50 frames (2 s) including overlaps of 25 frames (1 s) based on the result in Supplementary Fig. 1b. Regarding the cutoff frequency, we removed the DMD modes over 2 Hz based on the result in Supplementary Fig. 1c. Next, we describe the detailed computation of the feature vectors in classification using the Graph DMD modes.

Feature vectors using Graph DMD modes.   We first adjusted the computed DMD modes. One is a normalisation within the DMD modes such that the maximum absolute value of the elements is 1 because the amplitude of the modes depends on that of the time dynamics. Second is symmetrisation. Since the computed Graph DMD modes were different from symmetric matrices in a precise sense (although the ideal modes are symmetric matrices), we added the modes to the transposed modes and divided them by two. Finally, to evaluate the global complex behaviours, we averaged the DMD modes for all valid sliding windows.

Classification.   After Graph DMD, we perform classification in three different approaches. For details, see Supplementary Text 1. The first two approaches create feature vectors and the third automatically extracts the features via a neural network approach. The final two approaches computed graph features using existing methods: the second is graph Laplacian eigenvalues[39] as a baseline feature of a graph[45] (denoted GDMD Laplacian) and the third is deep graph convolutional neural networks[46] (denoted GDMD GCN) as a recent promising method for classifying the graph data. For all classification methods with the exception of the neural network approaches, we adopted logistic regression as a simple linear binary classification model.

As comparable methods for classifying graph sequence data, we adopted four methods. The first is a method using vectorised basic DMD modes[47] as a baseline of DMD approaches. The second is the Koopman spectral kernel[40] using DMD with reproducing kernels[41] as the existing method[38] for classifying the collective motion dynamics. The third is a hand-crafted feature as a simple baseline method. Fourth is an advanced neural network approach for classifying graph sequence data called spatio-temporal GCN[36] as a recent promising end-to-end method. For the detailed setups, see Supplementary Text 1.

Statistical analysis.   For an accurate evaluation of classification performance from the multiple viewpoints, in addition to accuracy, AUC and F-measure were calculated to compare the classification performance. AUC is based on the ROC curve, which is generated by plotting a cumulative distribution function of the true-positive rate with respect to the false-positive rate. Then, AUC takes various decision thresholds into consideration. Next, recall and precision rate were computed for F-measure. Recall rate is defined as the ratio of the sum of true positives and true negatives to the number of true positives (the true-positive rate), and the precision rate is defined as the ratio of the sum of true positives and true negatives to false positives. The trade-off curve between recall and precision was created using the cumulative distribution function. To evaluate the trade-off, the F-measure was calculated as follows: $(2 \times \text{precision rate} \times \text{recall rate})/(\text{precision rate} + \text{recall rate})$.

For investigating the robustness of the classification results, we repeated test sessions for logistic regression five times using different test sets in analogy to five-fold cross-validation (i.e. classified 25 times). To compare the various methods, since the hypothesis of homogeneity of variances between methods was rejected with Levene's test, the Kruskal-Wallis nonparametric tests were performed. As the post-hoc comparison, Wilcoxon rank sum

test with Bonferroni correction was used within the factor where a significant effect in Kruskal-Wallis test was found. Since comparisons were only performed between GDMD spectrum and others based on our hypothesis, p-value was multiplied by six. We used r values as the effect size for Wilcoxon rank sum test.

We adopted logistic regression for simply investigating the effect of each element of the feature vector. We calculated the odds ratio (related to effect size), its p-value and 95% confidence interval. For all statistical calculations, $p < 0.05$ was considered significant. All statistical analyses were performed using the MATLAB 2018a Statistics and Machine Learning Toolbox (The MathWorks, Inc., MA, USA).

References

1. Vespignani, A. Predicting the behavior of techno-social systems. *Science* **325**, 425–428 (2009).
2. Holme, P., Edling, C. R. & Liljeros, F. Structure and time evolution of an internet dating community. *Social Networks* **26**, 155–174 (2004).
3. Read, J. M., Eames, K. T. & Edmunds, W. J. Dynamic social networks and the implications for the spread of infectious disease. *Journal of The Royal Society Interface* **5**, 1001–1007 (2008).
4. Bansal, S., Read, J., Pourbohloul, B. & Meyers, L. A. The dynamic nature of contact networks in infectious disease epidemiology. *Journal of Biological Dynamics* **4**, 478–489 (2010).
5. Bullmore, E. & Sporns, O. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience* **10**, 186–198 (2009).
6. Breakspear, M. Dynamic models of large-scale brain activity. *Nature Neuroscience* **20**, 340 (2017).
7. Colizza, V., Pastor-Satorras, R. & Vespignani, A. Reaction–diffusion processes and metapopulation models in heterogeneous networks. *Nature Physics* **3**, 276 (2007).
8. Holme, P. & Saramäki, J. Temporal networks. *Physics Reports* **519**, 97–125 (2012).
9. Newman, M. E. & Girvan, M. Finding and evaluating community structure in networks. *Physical Review E* **69**, 026113 (2004).
10. Grady, D., Thiemann, C. & Brockmann, D. Robust classification of salient links in complex networks. *Nature Communications* **3**, 864 (2012).
11. Choi, M., Shi, J., Zhu, Y., Yang, R. & Cho, K.-H. Network dynamics-based cancer panel stratification for systemic prediction of anticancer drug response. *Nature Communications* **8**, 1940 (2017).
12. Rosenthal, S. B., Twomey, C. R., Hartnett, A. T., Wu, H. S. & Couzin, I. D. Revealing the hidden networks of interaction in mobile animal groups allows prediction of complex behavioral contagion. *Proceedings of the National Academy of Sciences* **112**, 4690–4695 (2015).
13. Wang, L., Shi, H. & Sun, Y.-X. Induced synchronization of a mobile agent network by phase locking. *Physical Review E* **82**, 046222 (2010).
14. Zhou, J. *et al.* Connection adaption for control of networked mobile chaotic agents. *Scientific Reports* **7**, 16069 (2017).
15. Croft, D. P., Krause, J. & James, R. Social networks in the guppy (poecilia reticulata). *Proceedings of the Royal Society of London B: Biological Sciences* **271**, S516–S519 (2004).
16. Tanner, H. Flocking with obstacle avoidance in switching networks of interconnected vehicles. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA'04)*, vol. 3, 3006–3011 (IEEE, 2004).
17. Bode, N. W., Wood, A. J. & Franks, D. W. Social networks and models for collective motion in animals. *Behavioral Ecology and Sociobiology* **65**, 117–130 (2011).
18. Mortveit, H. & Reidys, C. *An introduction to sequential dynamical systems* (Springer Science & Business Media, 2007).
19. Wu, C. W. Synchronization in networks of nonlinear dynamical systems coupled via a directed graph. *Nonlinearity* **18**, 1057–1064 (2005).
20. Cliff, O. M., Prokopenko, M. & Fitch, R. An information criterion for inferring coupling of distributed dynamical systems. *Frontiers in Robotics and AI* **3**, 71 (2016).
21. Hutchins, E. The technology of team navigation. *Intellectual teamwork: Social and technological foundations of cooperative work* **1**, 191–220 (1990).
22. Fujii, K. *et al.* Resilient help to switch and overlap hierarchical subsystems in a small human group. *Scientific Reports* **6** (2016).
23. Aoki, I. A simulation study on the schooling mechanism in fish. *Bulletin of the Japanese Society of Scientific Fisheries* **48**, 1081–1088 (1982).
24. Helbing, D. & Molnar, P. Social force model for pedestrian dynamics. *Physical Review E* **51**, 4282 (1995).
25. Centola, D. & Macy, M. Complex contagions and the weakness of long ties. *American Journal of Sociology* **113**, 702–734 (2007).
26. Idé, T. & Kashima, H. Eigenspace-based anomaly detection in computer systems. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'04)*, 440–449 (ACM, 2004).
27. Koopman, B. O. Hamiltonian systems and transformation in hilbert space. *Proceedings of the National Academy of Sciences* **17**, 315–318 (1931).
28. Mezić, I. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics* **41**, 309–325 (2005).
29. Rowley, C. W., Mezić, I., Bagheri, S., Schlatter, P. & Henningson, D. S. Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics* **641**, 115–127 (2009).
30. Schmid, P. J. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics* **656**, 5–28 (2010).
31. Proctor, J. L. & Eckhoff, P. A. Discovering dynamic patterns from infectious disease data using dynamic mode decomposition. *International Health* **7**, 139–145 (2015).
32. Brunton, B. W., Johnson, L. A., Ojemann, J. G. & Kutz, J. N. Extracting spatial-temporal coherent patterns in large-scale neural recordings using dynamic mode decomposition. *Journal of Neuroscience Methods* **258**, 1–15 (2016).
33. Fujii, K. & Kawahara, Y. Dynamic mode decomposition in vector-valued reproducing kernel hilbert spaces for extracting dynamical structure among observables. *Neural Networks* **117**, 94–103 (2019).
34. Delvenne, J.-C., Lambiotte, R. & Rocha, L. E. Diffusion on networked systems is a question of time or structure. *Nature Communications* **6**, 7366 (2015).
35. Nitzan, M., Casadiego, J. & Timme, M. Revealing physical interaction networks from statistics of collective dynamics. *Science Advances* **3**, e1600396 (2017).
36. Yan, S., Xiong, Y. & Lin, D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 7444–7452 (2018).
37. Yu, B., Yin, H. & Zhu, Z. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 3634–3640 (AAAI Press, 2018).
38. Fujii, K., Kawasaki, T., Inaba, Y. & Kawahara, Y. Prediction and classification in equation-free collective motion dynamics. *Plos Computational Biology* **14**, e1006545 (2018).

39. Chung, F. R. *Spectral graph theory*. 92 (American Mathematical Soc., 1997).
40. Fujii, K., Inaba, Y. & Kawahara, Y. Koopman spectral kernels for comparing complex dynamics: Application to multiagent sport plays. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD'17)*, 127–139 (Springer, 2017).
41. Kawahara, Y. Dynamic mode decomposition with reproducing kernels for koopman spectral analysis. In *Advances in Neural Information Processing Systems 29*, 911–919 (2016).
42. Klus, S., Gelß, P., Peitz, S. & Schütte, C. Tensor-based dynamic mode decomposition. *Nonlinearity* **31**, 3359 (2018).
43. Oseledets, I. V. Tensor-train decomposition. *SIAM Journal on Scientific Computing* **33**, 2295–2317 (2011).
44. Hojo, M., Fujii, K., Inaba, Y., Motoyasu, Y. & Kawahara, Y. Automatically recognizing strategic cooperative behaviors in various situations of a team sport. *Plos One* **13**, e0209247 (2018).
45. de Lara, N. & Pineau, E. A simple baseline algorithm for graph classification. *arXiv preprint arXiv:1810.09155* (2018).
46. Zhang, M., Cui, Z., Neumann, M. & Chen, Y. An end-to-end deep learning architecture for graph classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 4438–4445 (2018).
47. Tu, J. H., Rowley, C. W., Luchtenburg, D. M., Brunton, S. L. & Kutz, J. N. On dynamic mode decomposition: Theory and applications. *Journal of Computational Dynamics* **1**, 391–421 (2014).
48. Takeishi, N., Kawahara, Y. & Yairi, T. Learning koopman invariant subspaces for dynamic mode decomposition. In *Advances in Neural Information Processing Systems 30*, 1130–1140 (2017).
49. Fujii, K., Yamashita, D., Yoshioka, S., Isaka, T. & Kouzaki, M. Strategies for defending a dribbler: categorisation of three defensive patterns in 1-on-1 basketball. *Sports Biomechanics* **13**, 204–214 (2014).
50. Fujii, K., Isaka, T., Kouzaki, M. & Yamamoto, Y. Mutual and asynchronous anticipation and action in sports as globally competitive and locally coordinative dynamics. *Scientific Reports* **5** (2015).
51. Fujii, K., Yoshioka, S., Isaka, T. & Kouzaki, M. The preparatory state of ground reaction forces in defending against a dribbler in a basketball 1-on-1 dribble subphase. *Sports Biomechanics* **14**, 28–44 (2015).
52. Hojo, M., Fujii, K. & Kawahara, Y. Analysis of factors predicting who obtains a ball in basketball rebounding situations. *International Journal of Performance Analysis in Sport* 1–14 (2019).
53. Fujii, K. & Kawahara, Y. Supervised dynamic mode decomposition via multitask learning. *Pattern Recognition Letters* **122**, 7–13 (2019).
54. Kutz, J. N., Brunton, S. L., Brunton, B. W. & Proctor, J. L. *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems* (SIAM, 2016).
55. Lusch, B., Kutz, J. N. & Brunton, S. L. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications* **9**, 4950 (2018).
56. Álvarez, M. A. *et al.* Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning* **4**, 195–266 (2012).
57. Fujii, K., Takeishi, N., Kibushi, B., Kouzaki, M. & Kawahara, Y. Data-driven spectral analysis for coordinative structures in periodic human locomotion. *Scientific Reports* **9**, 1–14 (2019).

Acknowledgements

Author contributions

K.F. conceived the study. K.F. developed and implemented the software with N.T. and Y.K. guidance. K.F., M.H. and Y.I. conducted the experiments to obtain the data. K.F. analysed the results. All authors wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41598-020-58064-w.

Correspondence and requests for materials should be addressed to K.F.

Reprints and permissions information is available at www.nature.com/reprints.

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