IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 May 2024, accepted 12 July 2024, date of publication 17 July 2024, date of current version 25 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3429529

RESEARCH ARTICLE

Feature Extraction Method Using Lag Operation for Sub-Grouped Multidimensional Time Series Data

YUYA OKADOME[1] **AND YUTAKA NAKAMURA**[2]

[1]Faculty of Engineering, Tokyo University of Science, Katsushika, Tokyo 125-8585, Japan
[2]RIKEN Information Research and Development and Strategy Headquarters, Advanced Telecommunications Research Institute International, Seika, Soraku, Kyoto 619-0288, Japan

Corresponding author: Yuya Okadome (okadome@rs.tus.ac.jp)

This work was supported by Japan Science and Technology Agency (JST) Moonshot R&D Grant Number JPMJMS2011 (Development of Semi-autonomous Cybanetic Avatar) and Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Numbers (19H05693 and 23K16977).

ABSTRACT Systems in the real world often consist of multiple subsystems interacting with each other, for example, the musculoskeletal system or human-human interaction. The measurement of temporal changes in these systems involves a multidimensional time series. This study introduces a novel framework for extracting features from multidimensional time series data with a group structure using self-supervised learning techniques. Specifically, we use a ''lag operation,'' which is a temporal shifting operation applied to the features of a certain group. We propose a self-supervised learning method for a neural network model that uses the data automatically generated by the lag operation and its corresponding operation labels to capture and quantify the interaction between groups. Upon completion of the training process, the representation space is obtained with the expectation that it will capture timing-dependent features within its boundaries. We define and calculate the interaction score, R-score, on the obtained space. To validate our approach, we apply the proposed methodology to an artificial oscillator and approximately 4 hours of conversational data to evaluate the R-score properties. From the results of the artificial data, the R-score increases when the connection between the groups is large. From the high R-score region of the representation space of the conversation data, we extract the data that contain social behaviors such as ''eye contact,'' ''turn-taking,'' and ''smiling,'' which are related to the interaction between the participants. The experimental results suggest that the proposed method can obtain a representation space for time series data with a group structure.

INDEX TERMS Feature extraction, multi-dimensional time series data, deep learning, self-supervised learning.

I. INTRODUCTION

We measured a phenomenon that occurs in the real world, which may have a hierarchical structure. The elements of these measurements can be divided into several groups [1], [2], [3], [4]. In other words, each element in a group in the measurement interacts within the group, and the groups interact with each other. We considered a pre-training method for multidimensional time series data with a group structure affecting inter- and intra-group interaction.

The associate editor coordinating the review of this manuscript and approving it for publication was Tao Huang.

We assumed that our target system depended on the previous context with some parameters Θ.

$$P(\boldsymbol{x}(t)|\boldsymbol{x}(t-1), \boldsymbol{x}(t-2), \ldots; \Theta).$$

The interaction term v was defined for each group included in the system.

$$P^J(\boldsymbol{x}_{i\in J}(t)) = P^J(\boldsymbol{x}_{i\in J}(t)|\boldsymbol{x}_{i\in J}(t-1), \ldots, vg(X^{\neg J}); \Theta).$$

where $\boldsymbol{x}, J$, and g are the features, index set of the group, and function which governs the interaction among groups, respectively. The strength of the interaction is determined by

the magnitude of v. For simplicity, partial observability [5] was ignored, and the range of past information was defined by the time window in practice; for example, 10 time steps. Our aim was to model multidimensional time series data for several groups of elements in the measurement; the elements in a group are tightly coupled (highly correlated), and relatively weak interactions exist among the groups.

An example is the measurement of human motion using motion capture systems. Each musculoskeletal element, such as an arm or a leg, is composed of multiple joints that are interlocked internally and interact with each other at the arm and leg levels [1], [2]. Another example is the measurement of human-human interactions. Because each participant's reaction affects the others [3], [4], the probability of gesture expression at a certain time depends on the conversation partner's behavior, that is, the probability of the behavior is defined as the conditional probability.

Some studies have proposed feature extraction based on a deep neural network from time series data [6], [7], [8], [9]. Self-supervised learning is a widely used method, which employs a pair of sampled time series data for pretraining [6]. The hidden state of a recurrent neural network, for example, a long-short term memory-based auto-encoder model, is used for anomaly detection [7]. The self-supervised learning-based masked input that is obtained by deleting and reconstructing a certain region of time series data, is developed for modeling multi-rate time series data [8]. The transformer is also useful for extracting features from time-series data [9]. However, most studies do not discuss the group structure of multidimensional time-series data.

In this study, we proposed a self-supervised learning method based on a lag operation, in which the time of each group was shifted. It was assumed that the group structure was known in advance. The variables in the different groups, x_i and $x_{i'}$, do not always directly affect each other, and the strength of the effect is time-varying. For example, in human-human communication, the intensity of the interaction would be different between excitement and quiet situations.

By applying a lag operation to the time series data, the learned model constructs a distinctive feature space and extracts data, including group interactions, from a specific region in the space. A deep neural network model learns to predict the time shift and the representation space is obtained by projecting data using the learned network. The score of each data point is calculated based on kernel density estimation [10] to determine the location of each lag-operated data point in the learned representation space. This evaluation value was used as the criterion for data extraction.

We applied our model to the feature extraction of artificial phase oscillators and human behavior during conversations. In the experiment using the artificial oscillator, the calculated score changed according to the connection strength of the intergroup. For the actual data, which were the participants' behavior during the dyadic conversation, synchronized behaviors such as nodding, smiling, and others, were included in the data with high scores. These results suggest that the proposed model could extract the features of time series data with a group-like structure.

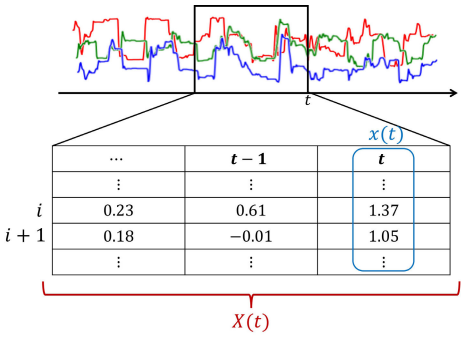


FIGURE 1. Relationship between $x(t)$ and $X(t)$. i is the index of the feature.

II. RELATED WORKS

A feature extraction method for multidimensional time series data based on deep neural networks was developed. The features of the time series data were expressed as latent variables using sequence-to-sequence architecture [11], in which the encoder-decoder architecture was constructed using long short-term memory [12]. The forecasting performance of the time series data could be improved by concatenating the hidden states of the encoder for each type of time series data (e.g., item sales and weather) [13]. An algorithm based on a convolutional neural network was used to extract features from the time series data, and this approach was applied to the classification of the data [14]. The representation of the time series data could be obtained using a transformer [15] similar to language modeling [9]. While these studies show the features of multidimensional time series data for some tasks that can be extracted using a deep learning approach, the group structure of the observed values is not considered.

A self-supervised learning technique was developed to obtain representations from unannotated information. Self-supervised learning was mainly used for small amount of labeled data, and network model weights were obtained by pretraining using unannotated data and automatically generated labels. Self-supervised learning approaches apply transformations ϕ to the input image, such as rotating and flipping [16], [17], [18] and breaking down an image into puzzle-like pieces [19], [20]. A neural network is trained on the converted data and automatically generated labels, that is, the image processes that are applied and the correct position of the broken image patches.

Efficient feature extraction from unannotated data has been achieved using deep neural networks [18], [21]. In this study, we adopted the self-supervised learning technique because this method can construct a feature space using unannotated samples, which can be obtained at a lower cost than annotated samples. In the proposed framework, the neural network learned to infer the amount of time shift using training data to cope with the interaction between groups.

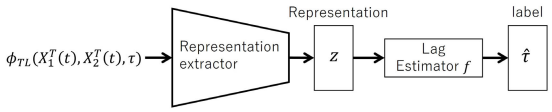


FIGURE 2. Input and output variables for self-supervised learning.

III. METHODS

This section describes the proposed learning framework for feature extraction from multidimensional time series data, which is trained using time shift labels. Part of the group behavior is assumed to interact, and the conversion of interaction data ϕ_{TL} is designed as a lag operation. The purpose of self-supervised learning is to extract features from the converted data.

A. PROBLEM SETTINGS AND NOTATIONS

This section describes the setting of the group data and notation of the variables. For simplicity, we make the following assumptions:

1 Two groups exist in the multidimensional time series data, wherein the features of the two at time t are $x_1(t)$ and $x_2(t)$.
2 The current state of our target time series data is affected by the finite length context, i.e., similar to the finite impulse response.

Based on these assumptions, T time-step features of the two groups are defined as $X_1^T(t) = [x_1(t-i)|i = 0,\ldots,T], X_2^T(t) = [x_2(t-i)|i = 0,\ldots,T]$. Fig. 1 illustrates the relationship between $x(t)$ and $X(t)$ at time t. The time indices of the features in each group $\cdot_L, \cdot_R$ remain consistent. In a typical supervised learning setting, a label is assigned to each sample by human annotators, and the neural network is trained using the annotated training data. However, annotation of the data is expensive, and it is preferable to reduce the annotation task.

In this study, we aimed to develop a framework for extracting the representation space for the temporal relevance of the two groups. We proposed an automatic data-synthesis procedure for a self-supervised method suitable for multidimensional and group data.

B. LAG OPERATION

This section describes the lag operation applied to the time series data. Corresponding to the time indices of both features, $X_1^T(t), X_2^T(t)$, the time-shifted feature is defined as $X_1^T(t), X_2^T(t+\tau)$. Thus, the lag operator ϕ_{TL} is expressed as follows:

$$\phi_{TL}(X_1^T(t), X_2^T(t), \tau) = \{X_1^T(t), X_2^T(t+\tau)\}, \tag{1}$$

where τ is the lag operation parameter, which is the amount of time lag, that is, the time shift. τ is sampled from the set of time shifts $\mathcal{T}$. Without loss of generality, X_2 is swapped with X_1 owing to temporal data symmetry.

Fig. 2 illustrates the input features, output variables, and estimation model. $\phi_{TL}(X_1^T(t), X_2^T(t), \tau)$ is the input to the model, and the model outputs the representation z. Feature z is input into the lag estimator, f, to estimate the shift $\hat{\tau}$, which indicates the amount of time shift.

C. LOSS FUNCTION

This section describes the loss functions used to train the feature extractor. $\tau \in \mathcal{T}$ is used as a label for the training process. To classify the number of lag operations, the following loss function,

$$L(z^p, \tau^B, \hat{\tau^B}) = \alpha L_c(\tau^B, \hat{\tau^B}) + \beta L_d(z^p, \tau^B), \tag{2}$$

was calculated. z^p, B, τ^B, and $\hat{\tau}^B$ are the representation, batch size, amount of lag operation for each data, and estimated amount of time shift, respectively. α and β are the constant weights for each term. Furthermore, L_c and L_d are the classification losses used to estimate τ and the distance-based losses, respectively, to determine the placements of the representations.

L_c is defined as the cross-entropy loss

$$L_c(\tau, \hat{\tau}) = \frac{1}{b}\sum_{b=1}^{B}\sum_{i\in\mathcal{T}} p(\tau = i)\log(p(\hat{\tau} = i)) \tag{3}$$

to estimate the discrete label τ. To learn the distance of each feature, L_d is defined as the soft-nearest neighbor loss [22]

$$L_d(z^p, \tau^B) = -\frac{1}{b}\sum_{b=1}^{B}\log\left(\frac{\sum_{\substack{j\in 1\cdots B\\ j\neq b}} \exp^{-\frac{d(z_b^p, z_j^p)}{T}}}{\sum_{\substack{k\in 1\cdots B\\ k\neq b}} \exp^{-\frac{d(z_b^p, z_k^p)}{T}}}\right), \tag{4}$$

where T and $d(\cdot)$ are the temperature variable and distance functions, respectively. The L2-norm $d(x_b, x_j) = ||x_b - x_j||^2$ was used in this study. Representations with the same τ were placed to close the area in the representation space by L_d. By employing both L_c and L_d, classification and placement problems can be handled simultaneously. The development of a loss function for self-supervised learning with interaction data is an area of future research.

D. CALCULATION OF R-SCORE

This section describes the calculation method for the score of sliced data based on the learned network model. The probability density was estimated based on the learned representation space, specifically with respect to the amount of lag operation in each case. The density of the representation for each operation τ was then computed, and estimated using the kernel density estimation method [10]. The score was defined as the location of the lag-operated data in the learned representation space.

The set of data with τ was defined as $x^\tau = \{\phi_{TL}(X_1^T(t), X_2^T(t), \tau)|\tau \in \mathcal{T}, t = 1,\ldots,N\}$. The representation z^τ was extracted from x^τ, and the probability

density was calculated as follows:

$$K(z^\tau, h) = \frac{1}{Nh}\sum_{i=1}^{N} k\left(\frac{z^\tau - z_i^\tau}{h}\right), \tag{5}$$

where $k(\cdot)$ is the kernel density function. A Gaussian kernel function was used in this study. Parameters h and N are the bandwidths of the kernel function and the sample size of the dataset, respectively. The density ratio for z is calculated as

$$R(z, h, \mathcal{T}) = \frac{1}{|\mathcal{T}|}\sum_{i\in\mathcal{T}} \frac{K(z^i; z^i, h)}{\sum_{j\in\mathcal{T}} K(z^i; z^j, h)}. \tag{6}$$

This $R(\cdot)$ is used as the ''score'' of the representations. The R-score increases when the operational data $z^i, i \in \mathcal{T}$ are projected onto the region where the same operated data exist. However, the score was approximately 0.2 when z^i was projected onto the undistinguished region, that is, the region where all types of operated data existed. If unknown data are input or each operated data is projected into the wrong area, the R-score becomes zero. In this case, evaluating whether the data are ''good'' is difficult. The data with $R \approx 0.2$ are defined as ''low score'' data.

IV. EXPERIMENTS

The proposed feature extraction framework was applied to two multidimensional time series data: the artificial phase oscillator and human-human conversation data. In the artificial oscillator experiment, the temporal data were generated from two multidimensional oscillators with interaction terms. The data of another experiment were obtained from dyadic conversation data, and two participants' facial features, that is, the roll, pitch, and yaw of the head, facial action units, and audio features, were considered.

A. ARTIFICIAL DATA BY PHASE OSCILLATOR

In this experiment, we applied our method to model the artificial time series data and investigate the characteristics of the model. An artificial oscillator, which is a nonlinear cyclic signal, was used and two oscillators were connected. The connection strength of the oscillators was changed. The R-scores for the time series data obtained from the oscillators were calculated.

1) PHASE OSCILLATOR

We used Kuramoto's phase oscillator [23] to generate artificial data and determine whether the proposed model could extract features related to the phase difference. A phase oscillator is a model in which each state is entrained into a phase difference defined by its parameters. When two oscillators with each oscillator having a different basis frequency, are connected to the weight, the feature of the nonlinear time series data changes because the strength of entrainment changes according to the magnitude of the weight. The oscillators converge to the same cyclic signal if the weight value is large, and the perturbations have an independent frequency if the weight is small.

Two oscillators with different parameters were used in this experiment. The interaction between the outputs was controlled by a weight term. The state of the jth unit in the ith oscillator changes according to the following dynamics:

$$\begin{aligned}\theta_{i,j}(t+1) = \theta_{i,j}(t) &\\ + \Delta\{\omega_i + \sum_k \sin(\theta_{i,k}(t) - \theta_{i,j}(t) - w_{i,j,k}) &\\ + u_{i,j}(t)\}, &\end{aligned} \tag{7}$$

where $w_{i,j,k}$ and ω_i are the phase matrix and intrinsic angular velocity, respectively. $u_{i,j}$ is the input from the other oscillator, and the term is calculated as

$$\boldsymbol{u}_i(t) = \sum_l \boldsymbol{v}_{i,l} \sin(\boldsymbol{u}_l(t)). \tag{8}$$

In this study, the connection weight $\boldsymbol{v}$ was randomly initialized, and ω_1, ω_2, and the number of units were empirically set to 1.7, 2.0, and 5. The strength of the effect of the other oscillator was changed by $\boldsymbol{v}$, and the behaviors of the output signals $\sin(\theta_i)$ and $i \in \{1, 2\}$ were changed.

2) EXPERIMENTAL SETTINGS

The input features of the network model, architecture of the network, and training setting are described in this section. The connection weight between the oscillators was sampled from a uniform distribution, and a 500-step sequence was generated for each weight. The training data were constructed by conducting the sequence generation process 100 times, that is, w was generated 100 times. For the test data, w was set to 0.0, 0.25, 0.5, 0.75, and 1.0 and a 500-step sequence was generated for each weight.

The calculation of the R-score was affected not only by w but also by the length of the past information T. For example, the proposed model could not distinguish between the lag-operated data if the past information was insufficient. The time steps were set to $T = 10, 20$, and 50, and the representation space after training was investigated.

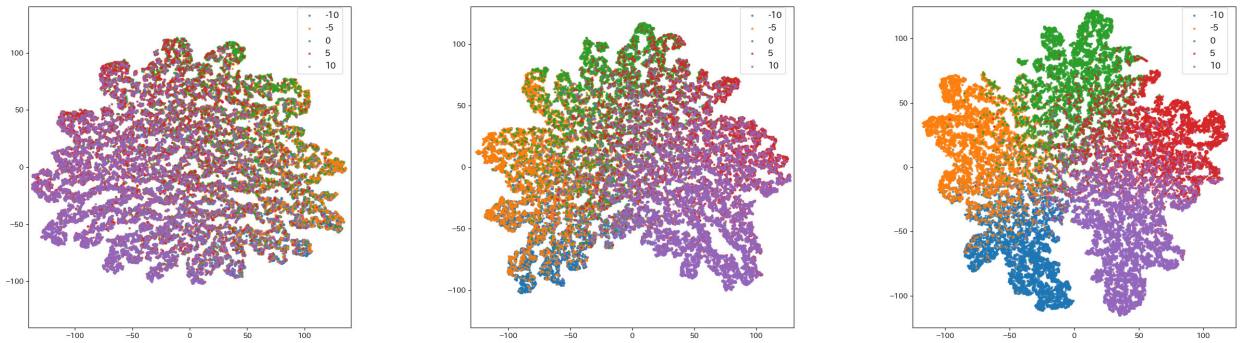
The architecture was a five-layer convolutional neural network with a kernel size of 3×3, and the features of each oscillator were handled as a two-channel image. The set containing the number of lag operations for self-supervised learning was $\mathcal{T} = [-10, -5, 0, 5, 10]$ steps. The constant variables in Equation 2 were set to $\alpha = 1.0, \beta = 0.2$. An Adam optimizer was used to learn the network, and the learning rate was set to 5×10^{-4}.

3) RESULTS

In this section, the relationship among the R-score, interaction term, and context length in the artificial data was investigated.

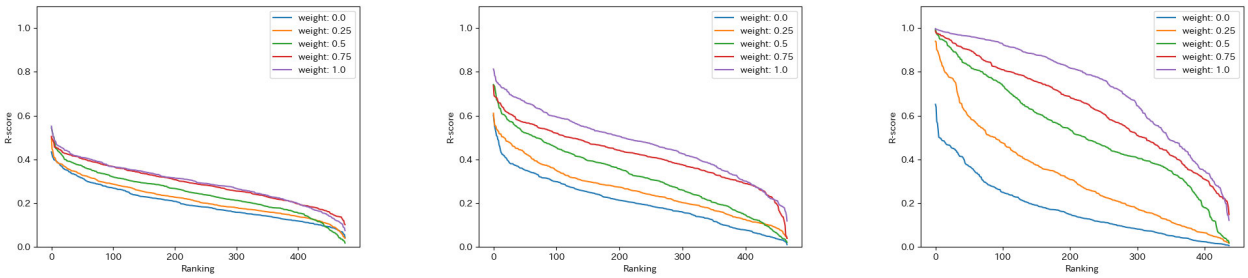
a: RESULTS OF REPRESENTATION SPACE

The learned representation spaces for each time step are presented in Fig. 3. When $T = 10$, most representations were not isolated, that is, all lag-operated data were mixed in



(a) $T = 10$ (b) $T = 20$ (c) $T = 50$

FIGURE 3. Example of compressed representation space with t-sne. Blue, orange, green, red, and purple dots represent the amount of lag operation $\tau = -10, -5, 0, 5, 10$ steps.



(a) $T = 10$ (b) $T = 20$ (c) $T = 50$

FIGURE 4. R-scores for each time step and connection weight. Blue, orange, green, red, and purple lines represent the sorted R-scores of $v = 0.0, 0.25, 0.5, 0.75, 1.0$ in descending order.

the representation space. Conversely, each lag-operated data was projected onto the difference region of the representation space when $T = 50$. Some representations were isolated when $T = 20$. To model the structure of multidimensional time series data, a certain amount of past information is necessary.

Even when $T = 50$, unseparated regions, such as $(x, y) = (0, 0)$, existed. It is important that data with a small $\boldsymbol{w}$ and before entrainment situation, are included in the dataset. For the separated region, the distances between the clusters of $\tau = 0$ and $\tau = \pm 5$ were smaller than those of $\tau = \pm 10$. Because data with the same amount of lag operation tends to project into the same region by the effect of the soft-nearest neighbor loss, other clusters with features similar to a certain cluster are expected to be placed close to each other.

b: RESULTS OF R-SCORE

Fig. 4 presents the sorted R-scores for each connection weight and time step. When $T = 10$, the maximum R-score for all weights was lower than 0.6, and the difference in the R-scores between the weights was insignificant. For $T = 20, 50$, the R-score increased from $w = 0.0$ to 1.0.

The R-score for $T = 50$ was largest, except in the case of $w = 0.0$. When $w = 0.75$ and 1.0, the reduction in the R-score did not drop dramatically, unlike in the other conditions. The feature extractor projects most of the lag-operated data onto the correct cluster. However, because data with small R-scores are included in the figure, each group is not entrained to all time steps, even if the weight is large. This result suggests that the proposed model can train multidimensional time series data with interaction effects between groups.

B. HUMAN-HUMAN CONVERSATION DATA

In the second experiment, we applied our proposed method to human-human conversation data [24]. The motions of the faces and voices were extracted from video clips and analyzed. Fig. 5 shows a schematic of the extracted features. The detailed preprocessing procedures are provided in Appendix A.

After training the model and score computation, data with a high R-score were extracted from the representation space to verify the possibility of extracting data with synchronization behavior, such as nodding. The extracted data were evaluated to determine whether synchronization behaviors were contained by checking the features of the data. In addition to the investigation, human impressions of the extracted videos scored using the R-score were evaluated (see Appendix B).

1) EXPERIMENTAL SETTINGS

The input features of the network model, architecture of the network, and training settings for conversation data are described in this section. From the collected 15 sessions in [25], 13 sessions were used as the training dataset and the remaining two were used as the test dataset. In the learned

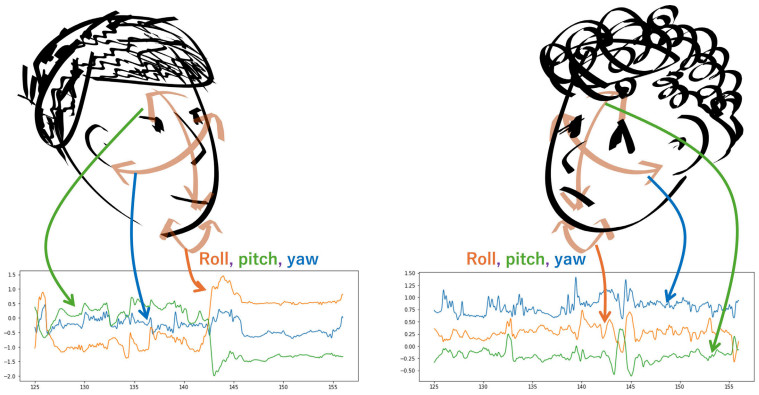


FIGURE 5. Schematic view of conversation data. Features during conversation are extracted.

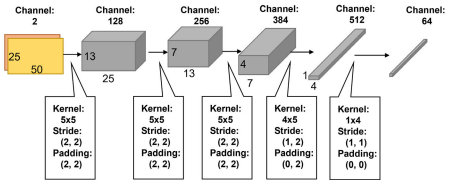


FIGURE 6. Network architecture of self-supervised learning.

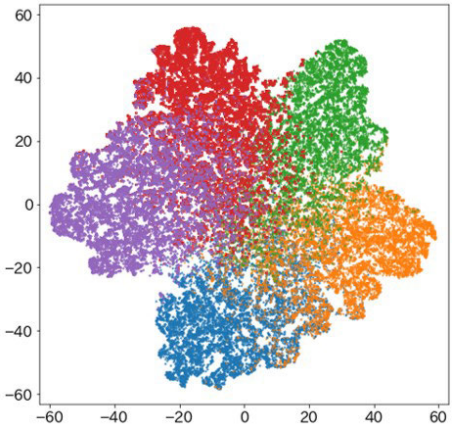


FIGURE 7. Example of compressed representation space with t-sne. Blue, orange, green, red, and purple dots represent the amount of lag operation $\tau = -1.0, -0.5, 0, 0.5, 1.0$, respectively.

model, the test data were the input and the representations were the output.

The length of the past information T was empirically set to $T = 50$, that is, a five-second context was used. To prevent the duplication of the information, the features were sampled every five frames. Therefore, the data in the training and test datasets were defined as $[X_1^{50}(t), X_2^{50}(t)|t = 50, 55, 60, \ldots]$. The numbers of samples for the training and test datasets were 30, 176, and 2, 467, respectively.

Fig. 6 illustrates the network architecture employed in this experiment. The architecture was a five-layer convolutional neural network, and the features of each participant were handled as two-channel images. The set containing the amount of lag operation for self-supervised learning is $\mathcal{T} = [-1s, -0.5s, 0s, 0.5s, 1s]$ with a maximum time shift of one second. The constant variables in Equation 2 were set to $\alpha = 1.0, \beta = 0.2$. An Adam optimizer was used to learn the network, and the learning rate was set to 5×10^{-4}. The feature extraction model was trained five times and the mean R-score was applied.

2) RESULTS OF THE BEHAVIOR EXTRACTION USING R-SCORE

In this section, the relationship between the R-score size and conversation data was evaluated. The representation space was obtained by learning the training dataset. The features after conversion, $\phi_{TL}(\cdot, \cdot, \tau)$ were expected to have time-dependent characteristics if isolated in the representation space. In contrast, the converted features with small behaviors were not separated, that is, each representation with τ was mixed. It is noteworthy that because the tendencies of the extracted data of the two test sessions were similar, the following results were discussed for the one test session.

a: REPRESENTATION SPACE

Fig. 7 shows an example of a representation space compressed using t-SNE [26] which is a visualization method for high-dimensional data. For τ, parts of the training data were separated. Significant unseparated data exist in the space, and these representations are placed at ''similar'' positions even if different τ were applied.

b: RESULTS OF FEATURE EXTRACTION

Fig. 9 shows the sorted R-scores for the dyadic conversation data. In human-human conversation, it is considered that many scenes are not synchronized because the reduction in the R-score is rapid.

Representations with high R-score (high-R) data and unseparated mixed (low-R) data with low R-scores were extracted. For low-Rs, if the input feature was projected onto the unseparated region, R became 0.2 from equation 6, that is, all kernel densities were expected to have the same value. Hence, high-R data are the data extracted from the highest R

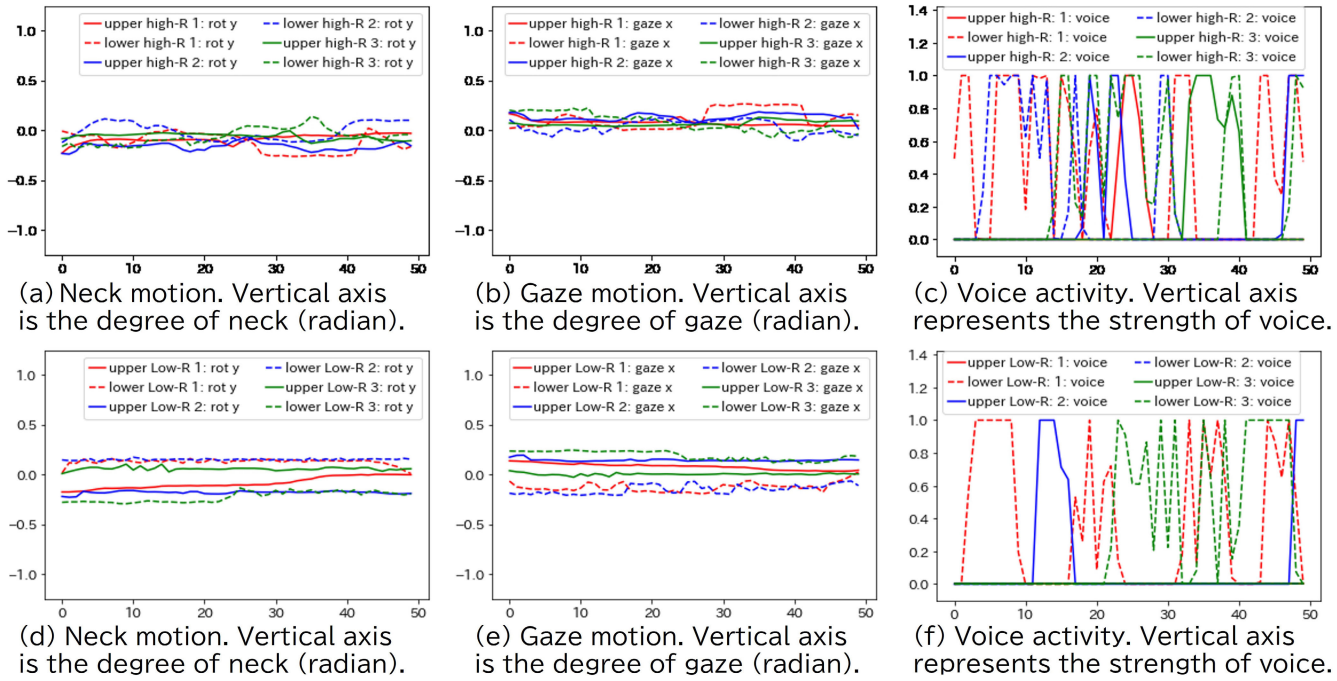


FIGURE 8. Examples of the extracted features. The placement of the upper and lower subject is the same as that in figure 12: (a), (b), and (c) are the features of three high-Rs; and (d), (e), and (f) are the features of three low-Rs. The horizontal axes represent the frames of the input data. The solid and dotted lines reflect the features of the "upper subject" and "lower subject," respectively. Red, blue, and green lines represent three high-Rs and three low-Rs, and solid and dotted lines with the same color are from the same data.

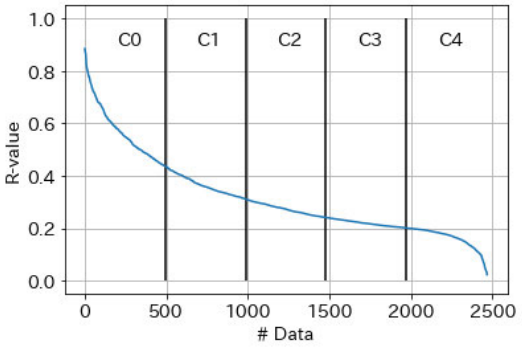


FIGURE 9. Sorted data based on the R-score. The clusters (C0, C1, C2, C3, and C4) were defined accordingly and used for the human evaluation shown in Appendix A. These clusters served as divided datasets, determined by their respective R-scores. Each cluster contains an equal number of data points.

in fig. 9, and low-R data are from approximately the 2000th data. If $R = 0.0$, the data were expected to project outside the representation space.

Fig. 8 shows the neck and gaze motions and VAD of the top three extracted high- and low-Rs. The motions in the figure are rotations around the y-axis, that is, yaw motions. For the neck motion, low-Rs indicated small changes, and the angular values were larger than the high-Rs. Because the subjects failed to anticipate beyond this point, the extracted data did not form interaction scenes.

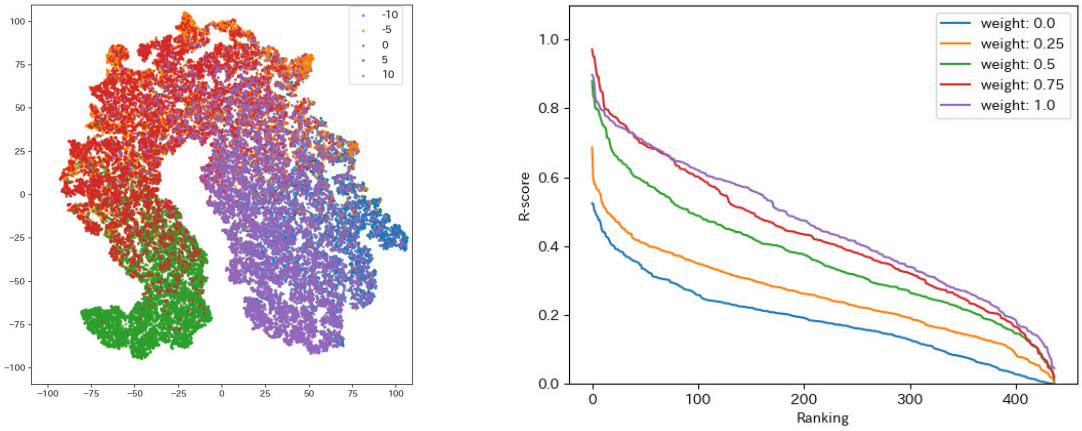
Regarding eye motion, the solid and dotted lines of the high-Rs demonstrate an intersection for the subjects, that is, making eye contact. In the case of low-Rs, eye contact did not occur because the lines were parallel. Eye contact is a synchronizing behavior because the feature extraction model easily detects the lag operation, that is, the R-score increases even though the mean value of the five training trials is calculated.

In VAD, turn-taking occured at high-Rs because the solid and dotted lines were alternatively activated. Only one subject continued to talk during low-Rs. The solid red line was activated at approximately 25 frames in low-R, whereas the dotted line was not activated until 40 frames. In comparison with the results of the gaze and neck motions, no synchronization behaviors were observed at low-Rs.

V. DISCUSSION

The proposed framework was characterized by a training method for a convolutional neural network and an R-score calculated in the representation space. From the distribution in the representation space, the experimental results demonstrated that the proposed framework could be used to extract the features of the multidimensional time series data.

In the proposed method, a simple convolutional neural network was used as a representation extractor. In particular in the image [18], [27] and language processing [28], [29] fields, the application of fine-tuning the fundamental model was developed for downstream tasks. Similar to these models, our proposed framework was expected to be a fundamental model for time series data such as human communication data ([25], in Japanese).



(a) Representation space (b) R-scores

FIGURE 10. Representation space and R-scores of each connection weight for phase oscillator data of three groups. The new oscillator with $\omega_3 = 2.5$ was added. The network architecture and learning setting, for example, the learning rate and the number of training and test data points, are the same as those used in the experiments with two groups' data. Blue, orange, green, red, and purple dots represent the amount of lag operation $\tau = -10, -5, 0, 5, 10$, respectively. Blue, orange, green, red, and purple lines represent the sorted R-scores of $w = 0.0, 0.25, 0.5, 0.75, 1.0$ in descending order, respectively.

In the experiment with dyadic conversation data, our method could extract data including synchronization behaviors, but the types of extracted behaviors have not been verified. It would also be interesting to evaluate the relationship between the R-score and social behavior (e.g., nodding, smiling, and eye contact). Such knowledge would be beneficial for the control rules of future communication robots.

The proposed method was applied to simple artificial data and small laboratory experimental data, and the experimental results suggest that a representation space can be extracted by the proposed method. We discuss the following points considered in time series analysis: real-time, contextual variations, and seasonal patterns. Our proposed method learns the representation space and the kernel density function of the space performed offline. In the real-time situation, observations during an experiment are projected onto the representation space, and then, the R-score is calculated by computing the value of the kernel density. While the calculation for the projection onto the feature space using the trained neural network is not large, the computational cost for the kernel density estimation increases as the number of samples $O(n)$. Therefore, to implement a real-time application, it is necessary to employ a computationally reasonable method for density estimation. The development of a framework for real-time processing using fast computational methods, such as approximations using mixed Gaussian distributions, is a future challenge.

To extract the features, there is a relationship between the time constant of change of the situation and the magnitude of time difference in the lag operation. Hence, large time lags and windows are considered necessary to evaluate contextual variation and seasonal patterns. However, if these situational changes alter the relationship between the observations, they will likely be projected to different feature points according to the situation by the proposed method.

Data synchronization is important because the proposed method is strongly dependent on time synchronization. The data were synchronized with high precision and could be processed appropriately because an omnidirectional camera was used in the experiment. However, if measurements are made using a large number of sensors to capture more complex phenomena, accurate sensor synchronization will be difficult to implemented. In such cases, a sensor network system must be used for practical applications.

For the proposed method, there are two types of scalabilities: the number and variety of data, and the group structure. Regarding the former, in the experiment of representation extraction from human-human conversation data, we used a limited amount of data (15 sessions and 20 minutes for each session). It is expected that the data contains smaller variety compared to that obtained from large-scale data collection. Such variety may affect the extracted features. Considering the extensions of the proposed method, such as the attribute-aware feature extraction method, it may be useful to handle such variations.

Regarding group structure, we confirmed that the proposed method works effectively for cases where there is only a two-group structure and the group composition is known in advance. Development of methods applicable to diverse phenomena, such as automatic extraction of group composition, will be considered for future study. As an evaluation of applicability to more complex phenomena, the results of an application to time series analysis of oscillators consisting of three groups are shown in figure 10. This is the result of applying the lag operation to one group out of the three; however, it can be seen that the distribution changes with different lags.

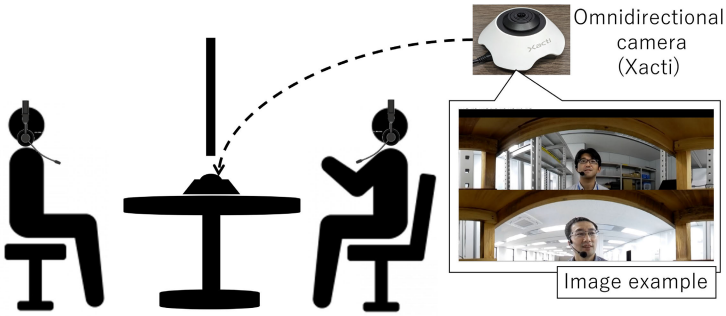


FIGURE 11. Schematic view of conversation data gathering. Each participant used the microphone, and individual cameras for each person are placed on the table.

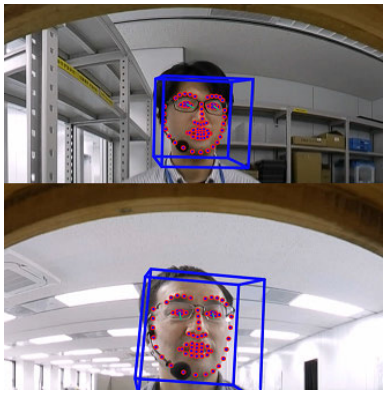


FIGURE 12. Example of face feature extraction.

Additionally, our proposed method involves some assumptions and limitations. The proposed method implicitly includes the following assumptions: 1) The target system ensures that the current feature is obtained depending on a finite length of the context. 2) The group structure is known. 3) Part of behaviors in each group is interacting with each other. From these assumptions, the following four limitations are considered: 1) A fixed time window is used. 2) The data that has three or more groups is not supposed in the framework. 3) The size of the time lag must be determined according to the task in advance. 4) Only a limited number and variation of data are used in the experiment. Relaxing these assumptions and limitations, such as by increasing the number of groups, can extend the application of the model.

VI. CONCLUSION

We proposed a self-supervised learning method based on a lag operation to model multidimensional time series data with a group structure. In this method, a lag operation, in which the time index of one subject was shifted, was applied to the time series data. The structured representation space was obtained by learning the label of the amount of the time shift. Using this feature extractor, each sample could be projected onto the representation space, and our method computed the R-score for each sample.

The proposed method was applied to artificial and actual datasets, that is, a phase oscillator and dyadic conversation. For the phase oscillator, we confirmed the change in the R-score according to the connection weight between each oscillator and the length of the time window. In the human-human conversation experiment, after learning using the lag-operated data, synchronization behaviors were selected based on the score criterion. These results indicated that the proposed approach could extract features from both artificial and actual time series data.

In the experiments, we modeled the multidimensional time series data with two groups. Situations involving a larger number of groups such as the three-party conversation, existed in the actual problem setting. The determination of the validity of the model in many groups will be a necessary task in future studies. In addition to the validity investigation, it is necessary to examine the availability of the extracted features because the proposed method is a feature extraction method, which involves a type of pretext task.

APPENDIX

A. PREPROCESSING PROCEDURES FOR HUMAN-HUMAN CONVERSATION DATA

1) DATA PROCESSING

The data gathering and feature extraction from the videos are described in this subsection. In this experiment, 15 sessions videos of dyadic conversation recorded [25] were used. Each session lasted approximately $10 - 20$ min and the total duration of each session was approximately 4 h. A video showing the facial information (Fig. 12) was recorded using an omnidirectional camera (Xacti CX-MT100). A dynamic microphone was placed near the mouth to observe the voice of each participant.

An omnidirectional camera was placed between two participants talking to each other, as shown in fig. 11, and the camera recorded the faces of the participants. Voice information was obtained using a headset microphone, and the microphone amplifier was adjusted such that only the voice of the person with a microphone was recorded.

The input features of the model were generated from the obtained data. From the video and audio data, three-dimensional face rotation (roll, pitch, and yaw) and the corresponding velocities, two-dimensional gaze rotation (x- and y-axes of an image) and the corresponding velocities, four-dimensional facial action unit (FAU), and voice activity

TABLE 1. Questionnaire items for the impression evaluation by crowd participants. "response, temporal, and attitude" in brackets are the groups of questionnaire items.

	Questionnaire items
Q1 (response)	Original: 提示された対話シーンにおいて、二人は盛り上がっているように感じましたか？ English: Were both participants engaged in an exciting conversation in the given dialogue scene?
Q2 (temporal)	Original: 提示された対話シーンにおいて、お互いの反応は噛み合っていましたか？ English: Were the reactions of both participants aligned with each other in the given dialogue scene?
Q3 (attitude)	Original: 提示された対話シーンにおいて、お互いの話を真剣に聞いていましたか？ English: Did both participants attentively listen to each other's conversations in the given dialogue scene?
Q4 (attitude)	Original: 提示された対話シーンにおいて、お互いに興味をもっていましたか？ English: Did both participants find each other interested in the given dialogue scene?
Q5 (response)	Original: 提示された対話シーンにおいて、話している人に反応を返していましたか？ English: Did both participants respond to the person speaking in the given dialogue scene?
Q6 (temporal)	Original: 提示された対話シーンにおいて、二人の振る舞いは調和していましたか？ English: Did the behavior of both participants harmonize in the given dialogue scene?
Q7 (temporal)	Original: 提示された対話シーンにおける、二人のテンポは良かったですか？ English: Was the tempo between both participants in the given dialogue scene good?
Q8 (response)	Original: 提示された対話シーンにおいて、二人は目を合わせていましたか？ English: Did both participants look each other in the eye well in the given dialogue scene?
Q9 (attitude)	Original: 提示された対話シーンにおいて、二人は考え込んでいましたか？ English: Did both participants appear lost in thought during the given dialogue scene?

TABLE 2. Result of Tukey's HSD test for C0 vs C1–C4. The statistics and *p*-values for the $\mathcal{G}1$ questionnaire group are described.

$\mathcal{G}1$	statistics	p-value
Q1: C0 vs C1	0.141	3.24×10^{-3}
Q1: C0 vs C2	0.283	1.11×10^{-11}
Q1: C0 vs C3	0.495	1.07×10^{-12}
Q1: C0 vs C4	0.310	1.12×10^{-12}
Q5: C0 vs C1	0.076	1.70×10^{-1}
Q5: C0 vs C2	0.222	9.00×10^{-10}
Q5: C0 vs C3	0.409	1.07×10^{-12}
Q5: C0 vs C4	0.263	1.28×10^{-12}
Q8: C0 vs C1	0.066	4.66×10^{-1}
Q8: C0 vs C2	0.126	1.46×10^{-2}
Q8: C0 vs C3	0.284	2.05×10^{-11}
Q8: C0 vs C4	0.258	1.65×10^{-9}

TABLE 3. Result of Tukey's HSD test for C0 vs C1–C4. The statistics and *p*-values for the $\mathcal{G}2$ questionnaire group are described.

$\mathcal{G}2$	statistics	p-value
Q2: C0 vs C1	0.085	5.53×10^{-2}
Q2: C0 vs C2	0.145	4.46×10^{-5}
Q2: C0 vs C3	0.304	1.07×10^{-12}
Q2: C0 vs C4	0.244	1.22×10^{-12}
Q6: C0 vs C1	0.054	4.99×10^{-1}
Q6: C0 vs C2	0.124	2.21×10^{-3}
Q6: C0 vs C3	0.254	1.63×10^{-12}
Q6: C0 vs C4	0.176	1.69×10^{-6}
Q7: C0 vs C1	0.071	3.24×10^{-1}
Q7: C0 vs C2	0.153	5.23×10^{-4}
Q7: C0 vs C3	0.353	1.07×10^{-12}
Q7: C0 vs C4	0.221	4.98×10^{-8}

detection (VAD) results were extracted. The video and audio sampling rates were set to 30 fps and 48 KHz, respectively.

2) FACE FEATURE EXTRACTION

By applying OpenFace [30] which is a software for face feature analysis in videos, the face position and features were estimated, as shown in Fig. 12. From the results of OpenFace, the face and gaze rotation, as well as the FAU, which is usually used for facial expression analysis, were obtained.

TABLE 4. Result of Tukey's HSD test for C0 vs C1–C4. The statistics and *p*-values for the $\mathcal{G}3$ questionnaire group are described.

$\mathcal{G}3$	statistics	p-value
Q3: C0 vs C1	0.141	$3.24 \times 10^{-}3$
Q3: C0 vs C2	0.283	$1.11 \times 10^{-}11$
Q3: C0 vs C3	0.495	$1.07 \times 10^{-}12$
Q3: C0 vs C4	0.310	$1.12 \times 10^{-}12$
Q4: C0 vs C1	0.076	$1.70 \times 10^{-}1$
Q4: C0 vs C2	0.222	$9.00 \times 10^{-}10$
Q4: C0 vs C3	0.409	$1.07 \times 10^{-}12$
Q4: C0 vs C4	0.263	$1.28 \times 10^{-}12$
Q9: C0 vs C1	0.066	$4.66 \times 10^{-}1$
Q9: C0 vs C2	0.126	$1.46 \times 10^{-}2$
Q9: C0 vs C3	0.284	$2.05 \times 10^{-}11$
Q9: C0 vs C4	0.258	$1.65 \times 10^{-}9$

3) VOICE ACTIVITY DETECTION

The voice activity was detected by distinguishing between voice and noise, including breathing and touching the microphone. To detect the voice activity of each participant, inaSpeechSegmenter [31] was applied to the gathered voices. inaSpeechSegmenter is a detection method based on the deep learning model, and its output is classified into "noise," "no energy," "music," and "speech." Appropriate sections of "speech" were selected and labeled as a result of VAD, and the power of "speech" was recorded.

4) COMBINING FEATURES

The input features of the self-supervised learning model were generated by combining the face motion, gaze motion, FAU, and VAD. To obtain the face-related features, each feature was downsampled from 30 to 10 fps to smooth the signals. The power of the voice was down-sampled

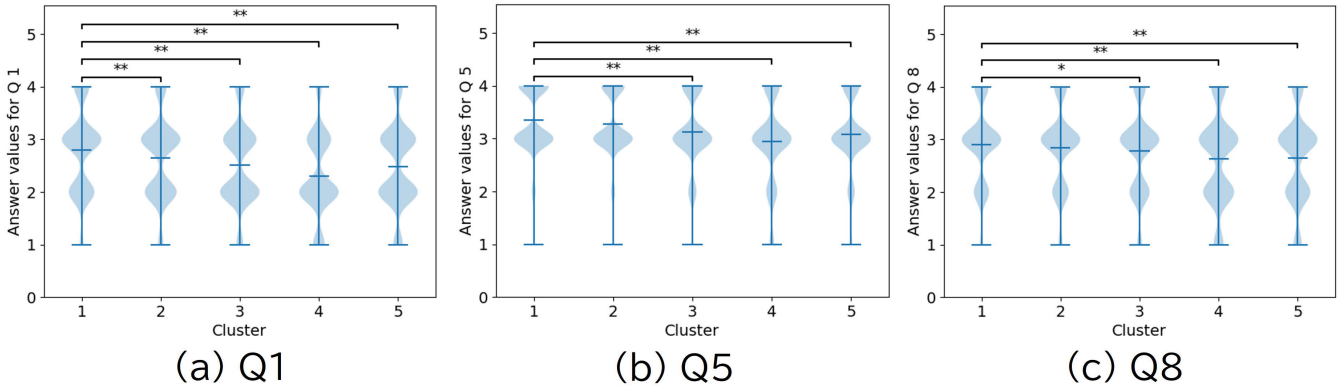


FIGURE 13. Answer values of $\mathcal{G}1$ (Q1, Q5, and Q8). The mean and results of Tukey's HSD test are illustrated. The horizontal and vertical axes are clusters (C0–C4) and evaluation values, respectively. Characters "*" and "**" show the significant differences with $p < 0.05$ and $p < 0.01$, respectively. Moreover, "+" represents significant trends ($p < 0.1$).

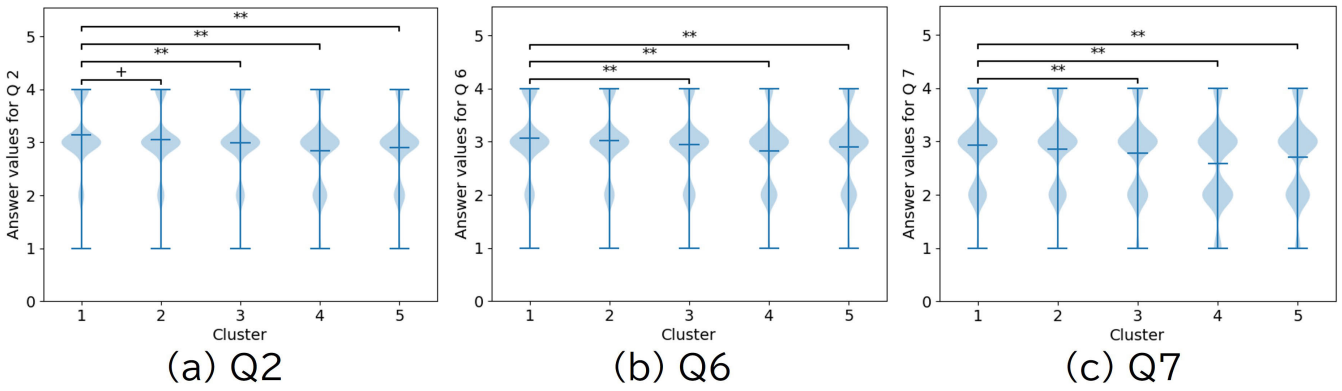


FIGURE 14. Answer values of $\mathcal{G}2$ (Q2, Q6, and Q7). The mean and results of Tukey's HSD test are illustrated. The horizontal and vertical axes are clusters (C0–C4) and evaluation values, respectively. Characters "*" and "**" show the significant differences with $p < 0.05$ and $p < 0.01$, respectively. Moreover, "+" represents significant trends ($p < 0.1$).

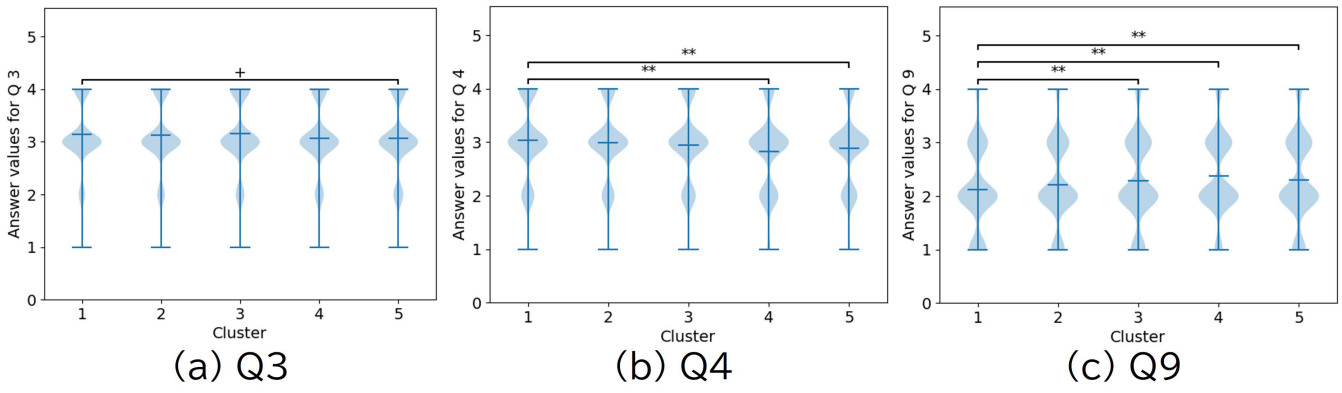


FIGURE 15. Answer values of $\mathcal{G}3$ (Q3, Q4, and Q9). The mean and results of Tukey's HSD test are illustrated. The horizontal and vertical axes are clusters (C0–C4) and evaluation values, respectively. Characters "*" and "**" show the significant differences with $p < 0.05$ and $p < 0.01$, respectively. Moreover, "+" represents significant trends ($p < 0.1$).

to $10Hz$ to calculate the maximum power for the past $48,000/10 = 4,800$ samples. These features were combined for each subject, and the input features $[x_1(t), x_2(t)]$ were generated for the learning model. Consequently, twenty-five-dimensional explanatory variables were obtained for each participant.

B. HUMAN IMPRESSION EVALUATION

This experiment evaluated the relevance between the R-score and human impression, that is, the meaning of the score was indirectly evaluated. Participants were advertised on a crowded platform through an agency that handled contracts and operations.

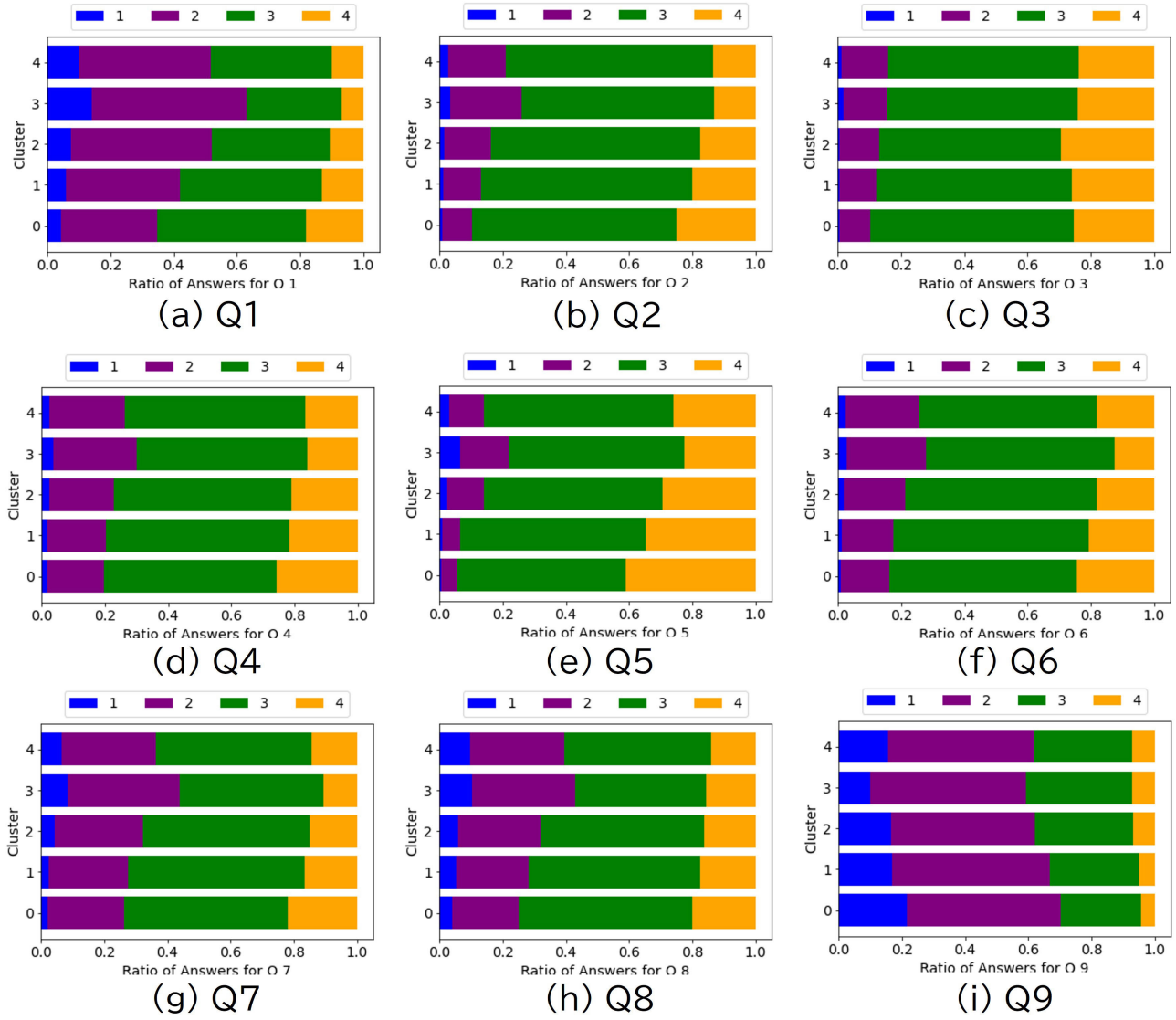


FIGURE 16. Band graphs of the ratio of evaluation values for each cluster. The horizontal and vertical axes are the ratios of each evaluation and clusters (C0–C4), respectively. Blue, purple, green, and yellow bands represent the evaluation values 1, 2, 3, and 4, respectively.

The results of previous experiments suggest that synchronization behaviors are obtained from data with a high R-score, and vice versa. All samples were divided into five clusters based on their R-scores, and samples from each cluster were presented to 200 participants. Each participant evaluated the videos by completing a questionnaire. In this experiment, the differences in human evaluations of the data extracted based on the R-scores were investigated.

1) EXPERIMENTAL SETTINGS

Fig. 9 shows the data sorted according to their R-scores. The sorted data were divided into five clusters as shown in the figure. The number of samples for each cluster was equal, that is, the amount of data in each cluster was $N/5$, where N is the size of the test data. In this experiment, the cluster with the highest R-score (C0) was compared to the other clusters (C1, C2, C3, and C4). Cluster C3 contained data with $R \approx 0.2$; therefore, C3 was considered a "bad" cluster.

For this experiment, 40 sets of videos were prepared, each containing 20 videos. For each set, four videos were sampled from each cluster, resulting in 160 videos being extracted from each cluster without duplication.

Each evaluator watched one set of videos, and the score for each watched video was recorded. For each set, five evaluators were assigned because of the score stability. Two hundred crowd participants were solicited to evaluate the videos.

Table 1 lists the questionnaire items used in this experiment. The original questionnaire items and English-translated version are presented. The questionnaire contained nine questions, mainly focusing on whether there was an exchange between the two individuals in each scene.

Each questionnaire item was divided into three groups

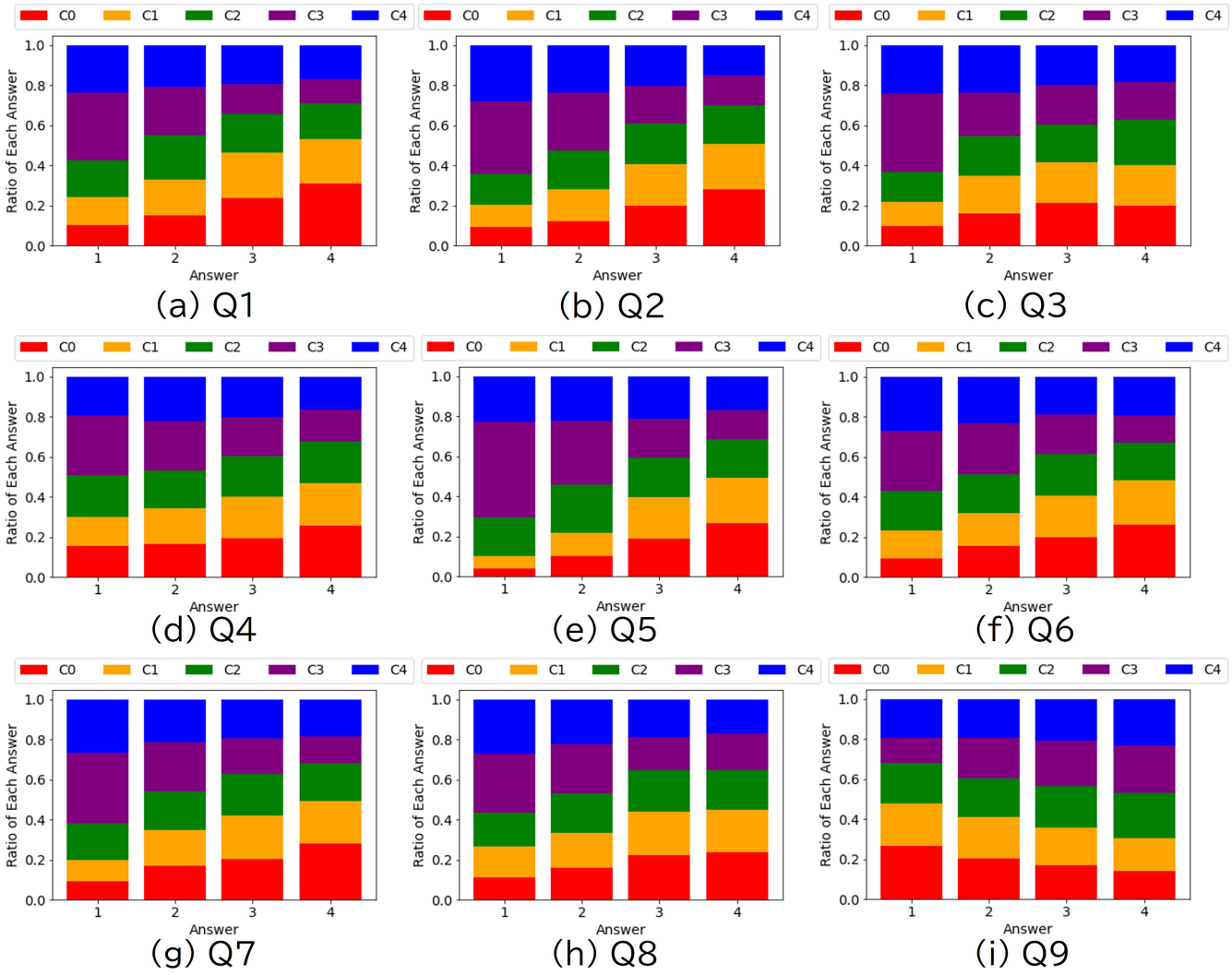


FIGURE 17. Cumulative bar graphs of the ratio of clusters for each evaluation value. The horizontal and vertical axes are evaluation values and the ratio of clusters. Red, yellow, green, purple, and blue bars are clusters C0, C1, C2, C3 and C4, respectively.

$\mathcal{G}1$: impression of conversation (Q1, Q5, and Q8),
$\mathcal{G}2$: tempo of conversation (Q2, Q6, and Q7),
$\mathcal{G}3$: perceived attitude of characters (Q3, Q4, and Q9).

Q9 is an inverted scale, that is, the evaluation value is expected to be high when the two participants are quiet and static (motionless). To prevent the concentration on "Neutral," the participant answered the score of the judge using a four-point Likert scale: 1) Strongly disagree; 2) Disagree; 3) Agree; 4) Strongly Agree.

2) RESULT OF STATISTICAL TEST

The following paragraphs present the results of Tukey's Honest Significant Difference (Tukey's HSD) test of the questionnaire groups, that is, the results of $\mathcal{G}1$, $\mathcal{G}2$ and $\mathcal{G}3$ between the cluster with the highest R-score (C0) and the other clusters.

a: RESULTS OF $\mathcal{G}1$

Fig. 13 summarizes the human evaluation of $\mathcal{G}1$. The differences between C0 and the other clusters were investigated using statistical tests. Table 2 presents the statistics and p-values of Tukey's HSD test.

For questionnaire item Q1, significant differences between C0 and the other clusters were observed. For Q5 and Q8, significant differences between C0 and certain clusters (C2, C3, C4) were observed, whereas no significant differences were observed between C0 and C1.

b: RESULTS OF $\mathcal{G}2$

Fig. 14 shows a summary of the human evaluations of $\mathcal{G}2$. The differences between C0 and the other clusters were investigated using statistical tests. Table 3 presents the statistics and p-values of Tukey's HSD test.

These results were similar to those of the response groups. For item Q2, significant differences between C0 and the other clusters were also observed. Regarding Q6 and Q7, significant differences between C0 and C2 and C3 and C4 were observed, whereas there were no significant differences between C0 and C1.

c: RESULTS OF $\mathcal{G}3$

Fig. 15 shows a summary of the human evaluation of $\mathcal{G}3$. Table 4 presents the statistics and p-values of Tukey's HSD test.

A significant difference between C0 and C4 in Q3 was observed. Based on the results of Q3, both participants talked to each other seriously. In Q9, which is an inverted scale, significant differences between C0 and C2 and C3 and C4 were observed, whereas no significant differences were observed between C0 and C1.

3) RESULTS OF ANSWER DISTRIBUTIONS

Fig. 16 shows a band graph of the ratios of the evaluation values for each cluster. The ratio of Agree (3) to Strongly Agree (4) was high in the cluster with a large R-score for Q1-Q8. Specifically, in Q1, the ratio of Agree to Strongly Agree for C0 was above 0.6, whereas that of C3 was approximately 0.4. Additionally, for Q9, C0 exhibited a high ratio of Strongly Disagree and Disagree, whereas C3 exhibited relatively low scores.

Fig. 17 presents a cumulative bar graph depicting the ratio of the clusters for each evaluation value. In Q1–Q8, the ratio of C0 increased as the evaluation value increased. In contrast, when the evaluation value decreased, the ratio of the data close to $R = 0.2$ (cluster C3: representing "Bad" data) increased. In C4, no large changes in the ratio across all questionnaire items were observed. This is because the "bad" and unknown data were included in C4.

These results suggest that "exciting" and "harmonizing" behaviors during the dyadic conversation scenes were extracted. Using a trained feature space with lag operations, extracting data that synchronizes with the behavior, even when evaluated by humans, was possible.

REFERENCES

[1] E. Todorov, "Direct cortical control of muscle activation in voluntary arm movements: A model," *Nature Neurosci.*, vol. 3, no. 4, pp. 391–398, Apr. 2000.

[2] W. Li and E. Todorov, "Iterative linear quadratic regulator design for nonlinear biological movement systems," in *Proc. 1st Int. Conf. Informat. Control, Autom. Robot.*, vol. 2, 2004, pp. 222–229.

[3] E. Delaherche, M. Chetouani, A. Mahdhaoui, C. Saint-Georges, S. Viaux, and D. Cohen, "Interpersonal synchrony: A survey of evaluation methods across disciplines," *IEEE Trans. Affect. Comput.*, vol. 3, no. 3, pp. 349–365, Jul. 2012.

[4] J. Kwon, K.-I. Ogawa, E. Ono, and Y. Miyake, "Detection of nonverbal synchronization through phase difference in human communication," *PLoS One*, vol. 10, no. 7, Jul. 2015, Art. no. e0133881.

[5] S. Thrun, "Probabilistic robotics," *Commun. ACM*, vol. 45, no. 3, pp. 52–57, 2002.

[6] K. Zhang, Q. Wen, C. Zhang, R. Cai, M. Jin, Y. Liu, J. Y. Zhang, Y. Liang, G. Pang, D. Song, and S. Pan, "Self-supervised learning for time series analysis: Taxonomy, progress, and prospects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 1, no. 1, pp. 1–20, Apr. 2024.

[7] H. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu, "Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT," *IEEE Sensors J.*, vol. 22, no. 23, pp. 22836–22849, Dec. 2022.

[8] J. Chen, P. Song, and C. Zhao, "Multi-scale self-supervised representation learning with temporal alignment for multi-rate time series modeling," *Pattern Recognit.*, vol. 145, Jan. 2024, Art. no. 109943.

[9] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A transformer-based framework for multivariate time series representation learning," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 2114–2124.

[10] C. M. Bishop and N. M. Nasrabadi, *Pattern Recognition and Machine Learning*, vol. 4. Cham, Switzerland: Springer, 2006.

[11] A. Sagheer and M. Kotb, "Time series forecasting of petroleum production using deep LSTM recurrent networks," *Neurocomputing*, vol. 323, pp. 203–213, Jan. 2019.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[13] Y. Okadome, W. Wei, R. Sakai, and T. Aizono, "Demand-prediction architecture for distribution businesses based on multiple RNNs with alternative weight update," in *Proc. 28th Int. Conf. Artif. Neural Netw.*, 2019, pp. 486–496.

[14] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *J. Syst. Eng. Electron.*, vol. 28, no. 1, pp. 162–169, Feb. 2017.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[16] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, Dec. 2020.

[17] Z. Feng, C. Xu, and D. Tao, "Self-supervised representation learning by rotation feature decoupling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10356–10366.

[18] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, vol. 119, 2020, pp. 1597–1607.

[19] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 69–84.

[20] P. Goyal, D. Mahajan, A. Gupta, and I. Misra, "Scaling and benchmarking self-supervised visual representation learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6390–6399.

[21] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, vol. 2, 2015, pp. 1–15.

[22] N. Frosst, N. Papernot, and G. Hinton, "Analyzing and improving representations with the soft nearest neighbor loss," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2012–2020.

[23] Y. Kuramoto, "Self-entrainment of a population of coupled non-linear oscillators," in *Proc. Int. Symp. Math. Problems Theor. Phys.*, 1975, pp. 420–422.

[24] Y. Okadome and Y. Nakamura, "Extracting feature space for synchronizing behavior in an interaction scene using unannotated data," in *Artificial Neural Networks and Machine Learning—ICANN*. Cham, Switzerland: Springer, 2023, pp. 209–219.

[25] Y. Okadome, K. Ata, H. Ishiguro, and Y. Nakamura, "Self-supervised learning method for behavior prediction during dialogue based on temporal consistency," *Trans. Japanese Soc. Artif. Intell.*, vol. 37, no. 6, p. 431, 2022.

[26] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–23, 2008.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[28] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi, "Self-instruct: Aligning language models with self-generated instructions," 2022, *arXiv:2212.10560*.

[29] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 1, pp. 5485–5551, 2020.
[30] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, "OpenFace 2.0: Facial behavior analysis toolkit," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit.*, May 2018, pp. 59–66.
[31] D. Doukhan, J. Carrive, F. Vallet, A. Larcher, and S. Meignier, "An open-source speaker gender detection framework for monitoring gender equality," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 5214–5218.

YUYA OKADOME received the B.E., M.E., and Ph.D. (Eng.) degree from Osaka University, Osaka, Japan, in 2011, 2013, and 2016, respectively. He is a Junior Associate Professor with the Faculty of Engineering, Tokyo University of Science, where he joined in 2023. In 2016, he joined Hitachi Ltd., as a Researcher, and moved to RIKEN Information Research and Development and Strategy Headquarters, as a Researcher, in 2021. His research interests include data analysis, machine learning, robotics, and intelligent systems.

YUTAKA NAKAMURA received the B.E. degree from Kyoto University, Japan, in 1999, and the M.E. and D.Eng. degrees from Nara Institute of Science and Technology, Japan, in 2001 and 2004, respectively. From 2004 to 2006, he was a Postdoctoral Researcher with Nara Institute of Science and Technology; and an Assistant Professor with the Graduate School of Engineering, Osaka University, Japan, from 2006 to 2010. From 2010 to 2020, he was with the Graduate School of Engineering Science, Osaka University, as an Assistant Professor (2011–2013) and an Associate Professor (2013–2020). He has been a Team Leader of Guardian Robot Project with the RIKEN Information Research and Development and Strategy Headquarters, RIKEN, Japan, since 2020. His research interests include reinforcement learning of locomotions, human–robot interaction and modeling, and human behavior during interaction.