2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
October 11-14, 2020. Toronto, Canada

Analyzing the Extent of Rapport in Groups of Triads Via Interactional Synchrony

Nicholas Wilkins
Google Inc.
Mountain View, CA USA
npwilkins@google.com

Ifeoma Nwogu
Department of Computer Science
Rochester Institute of Technology
Rochester, NY USA
ionvcs@rit.edu

***Abstract*—Research in social psychology has extensively shown that in cohesive groups, individuals often mirror each other's prosody, facial expressions, and body movements. This mirroring effect can help determine the level of comfort or the extent of engagement and genuine interest between two or more interlocutors. In this work, using an annotated dataset consisting of videos of three-person conversations, we aim to analyze the extent of rapport in each of the triadic groups. We generate behavioral curves from features extracted from the participants' face and body movements. These are the sampled time series signals resulting from their multimodal features. Next, the extents of synchrony are analyzed by aligning the behavioral curves of pairs of participants. The alignment tests show that basic correlation coefficient measures outperform more advanced curve matching techniques when used to estimate the similarities between multidimensional behavior curves. They also show that in this dataset, synchrony is better observed from facial expressions than body movements. For this reason, using facial action units, we show that an end-to-end recursive neural network (RNN) trained using a regression loss yields good results in predicting the extent of synchrony in small groups.**

***Index Terms*—Interactional Synchrony; Group Formation Task; Long-short term memory networks (LSTM);**

I. Introduction

Research has shown that up to two-thirds of human communication occurs via nonverbal channels such as gestures (or body movements), facial expressions, and affective speech prosody [1]. Therefore, in the last couple of decades, an extensive amount of computational work has been done in the research of analyzing facial expressions and head movements, and evaluating different prosodic cues for emotion recognition.

Studies have shown that interactional synchrony (the temporal coordination of micro-level social signals between two or more people communicating in a social setting) plays an important role in maintaining positive social relationships among people since it indicates increased affiliation, rapport and feelings of empathy [2]–[4]. Similarly, relevant studies recognize the role of synchrony in learning behaviors in work teams [5]. The literature indicates that synchrony is a hallmark of relationships, and is produced as a result of rapport [2], [6]. Developing and maintaining rapport is a critical component of successful interactions in different social settings.

This material is based upon work supported by the National Science Foundation under Grant No. 1846076.

Fig. 1. Sample video frame where three subjects are undergoing the group formation task

We therefore expect pairs or groups of individuals with high cohesion and established rapport to more strongly exhibit the *mirroring effect*, a form of interactional synchrony. The mirroring effect is a phenomenon which occurs when individuals mimic each other's behavior subconsciously to gain and keep rapport [7]. Thus, we expect the cohesion of a group to directly correspond to the extent of synchrony occurring within the group.

The ability to measure the extent of interactional synchrony among the members in a group can be used as a metric for cohesion or rapport within that group. The two main research questions we intend to address in this study, therefore, are:

1) In a conversation involving two or more interlocutors, can we computationally detect and measure the extent of interactional synchrony, and if so, how well do these computational measures compare with human perceptions of synchrony?
2) Compared with classical methods for evaluating time-series data, how well does our proposed approach perform?

To this end, when given two or more time series signals, our goal is to determine if they are interacting with each other, and if so, to what extent. The specific social context in which interactional synchrony is studied in this work is the *group formation task*, involving in how well informal

groups of strangers develop rapport in the presence/absence of alcohol consumption. The real world data set used for this work could potentially have some confounding factors as some of the participants were imbibing alcohol before/during the interaction. Due to the lack of associated labels[1], we could not account for such factors.

II. Related work on computationally analyzing interactional synchrony

While the notion of interactional synchrony has been studied extensively in the social psychology literature, much less work has been reported in the computational analysis literature on this subject. [8] presented an extensive survey of synchrony evaluation from a multidisciplinary perspective, focusing on psychologists' coding methods, non-computational evaluations and early machine learning techniques [8].

Synchronicity analysis has also been previously approached by using coupled hidden Markov models (cHMMs) [9], [10] to classify taichi movements, under the assumption that different parts of the body moving in taichi will be synchronous to each other. In a related study, Pentland and collaborators [11] performed a computational study which involved forms of interactional synchrony and group influences. Li *et al.* [12] presented a supervised model used to predict the outcomes of video-conferencing conversations in the context of new recruit negotiations [12]. Yu *et al.* [13] presented a technique to investigate interactive synchrony in facial expressions and showed using the Pearson's correlation measure, that synchrony features were effective at detecting deception. Hammal *et al.* [14] evaluated the temporal coordination of head movements in couples with a history of interpersonal violence, and Chu *et al.* [15] developed a search-based technique for unsupervised, accelerated, multi-synchrony detection.

Group cohesion analysis (not in the context of synchrony) has been previously approached by using using SVMs on extracted audiovisual features from group meeting videos labeled as having high or low cohesion by human annotators [16]. Bilakhia *et al.* [17] measured facial mimicry using long short-term memory (LSTM) and detected facial activities and Tervern *et al.* [18] evaluated head gesture mirroring using smart glasses. In this paper we describe our work on studying group rapport using both the traditional methods of measuring time series similarities as well as using a neural network model, specifically the LSTM.

III. Real-life Data: The Sayette Group Formation Task (GFT) Dataset

The data for this study was obtained from Girard *et al.* [19] which was drawn from a larger study on the impact of alcohol on group formation processes [20]. To obtain the dataset, individual subjects were recruited to study the impact of alcohol on how well individuals in a newly formed group could establish rapport with one another.

[1]Due to experimental protocol constraints, the collectors of the data could not provide labels of the persons in the groups that consumed alcohol

All participants in the study were previously unacquainted and met for the first time at the experiment. They were instructed to consume a beverage and then engage with two other study participants. The groups of interlocutors were made of up three such subjects who were engaged for about 30-40 minutes of unstructured interactions. For many participants we viewed, this was a rather awkward social setting, hence our considering this as a somewhat abnormal social setting. The data provided to us by [19] for this study *focused on a 1 minute portion of the entire video* where the collectors believed the participants in the group had become sufficiently acquainted with each other, *i.e.* it was not likely that the group would build additional rapport over the remaining course of the experiment.

Separate wall-mounted cameras faced each participant and another camera captured the overall group interaction, resulting in a total of four videos - one at the overall group level showing body movements and three at the individual subject level showing mainly the face. The dataset contained a total of 172,800 frames, with 1,800 frames for each of the 96 participants (32 groups of three).

There was a total of 96 participants in this dataset (42% female, 85% white) and they all consented to having their audiovisual data used in further experiments.

A. Human annotations of GFT data

In the absence of ground truth labels for the data, we requested five individual labelers to review all the videos in the dataset and provide an aggregate group synchrony score on a Likert scale, based on their perception of how well they thought the group was interacting. The labelers were instructed to also observe for synchrony across the entire group. The scores were in the range of 1 to 5, with 1 implying the group was completely unsynchronized, 5 being completely synchronized. The labelers were instructed to judge overall synchrony so that even if two people in the group were interacting well with each other, but not with the third person, the group could not receive a high score. Below are some guidelines used in assigning scores to the videos.

Score = 1: The people are meeting for the first time and not trying to form a group.
Score = 2: The people are meeting for the first time and are trying to form a group.
Score = 3: The people know each other.
Score = 4: The people know each other and are in the process of forming a group.
Score = 5: The people know each other well and already function well as a group.

Unfortunately, many statistical methods used to account for the subjectivity in the labeling, such as Cohen's Kappa or Correlations are designed to find disagreements between two labelers. In this work, we had 5 labelers each annotating or more videos so that each video had at least three labels.

To account for disagreements, we computed the total variances for the scores provided by all but one of the labelers (we did this five times), and removed the set of annotations that

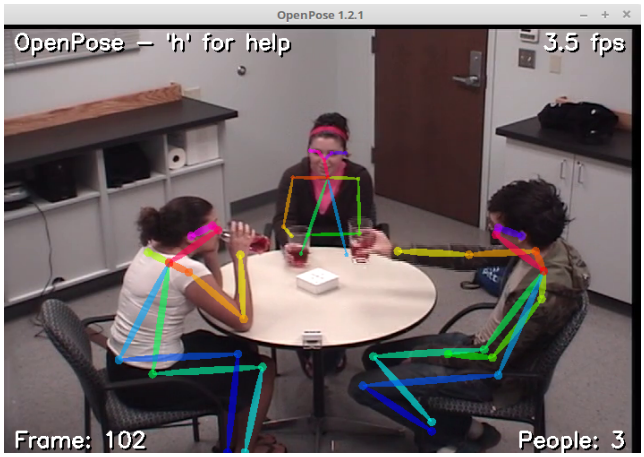

Fig. 2. OpenPose output on a sample frame from the GFT dataset.

caused the largest variance in the set. We also spot-checked the variances across each group and when this was larger than a preset threshold and had additional labelers re-annotate the video to break the discrepancy. This was done for only two groups in the entire dataset. The average scores obtained from the human labelers for each group was now considered as the human impression of rapport, the new gold-standard we used to train the network. We utilize human labelers in order to investigate whether our computational methods can behave similar to human perceptions of rapport.

B. Processing the GFT dataset

1) Facial Expressions: Each individual's facial expression was parsed using OpenFace [21], an open-source toolkit capable of facial landmark detection, head pose estimation, facial action unit recognition, and eye-gaze estimation. Action units (AUs) correspond to various muscle groups on the face and can range from being fully activated to not activated [22]. For each trio in the group, we took the three most active AUs in the dataset. Since the AUs are measured over time, we obtained a collection of behavioral signals (based on the face dynamics). The AUs of interest are AU6, AU7 and AU12 (AU 6 and AU 12 are associated with positive, happy emotions).

2) Body Joint Movements: Unlike AUs that fully characterize face movements and expressions, there is no similar well established set of elemental features that govern the movements and gestures made by the rest of the body. We therefore employ an open source toolkit for upper body joints location estimation.

OpenPose [23], [24] is a real-time multi-person open source toolkit used to detect human body parts in images and videos. Figure 2 shows an illustration of the OpenPose output when applied to a sample video from the Sayette GFT dataset.

OpenPose produces 2D locations and confidence values for 18 keypoints of a body model. For our dataset, we used only the seven keypoints from upper body parts as subjects in our dataset are sitting around the table and not moving around; keypoints from lower body, in our case, are not really relevant to study the synchrony among them, especially since the camera could not always have access to the lower body often occluded by the table or other participants.

3) Augmenting the GFT dataset: We took a sliding window corresponding to one second (30 frames) and denoted this as *look-back* temporal distance of the model we are analyzing. By sliding over the 1-minute videos with a stride of 1, we boosted the size of the data significantly. Similarly, as there was no logical ordering to how we selected the first, second or third persons, we rotated the orders of the subjects so that each set of sequences was processed six different times over six new configurations *i.e.* for three given sequences A, B, and C, we had the following configurations - ABC; ACB; BCA; BAC; CAB; CBA. Each 30-frame length window was labeled with the overall value that the annotators had ranked the extent of cohesion of the video of the drinking trio.

IV. Aligning Behavioral Curves

In this section, we describe three techniques for aligning behavioral curves to better study the phenomenon of interactional synchrony. We treated the AUs as one set of behavioral curves and the OpenPose joints as another set and using the various alignment techniques, our goals were twofold: (i) to determine the best technique for curve alignment using the annotations provided by human labelers; and (ii) to determine whether interactional synchrony was better evaluated at the facial expression level or at the body movements level.

The three techniques investigated are *correlation coefficient*, *dynamic time warping* and the *Riemannian elastic metric*.

A. Correlation coefficient - Baseline

Correlation is a statistical measure that indicates the extent to which two or more variables change together. A positive correlation indicates the extent to which those variables increase or decrease in parallel. The correlation coefficient is a statistical measure of the degree to which changes to the value of one variable predict change to the value of another.

If two signals are correlated but have a time delay or latency d between them, the latency can be accounted for in form of a sliding window of size d. The correlation coefficient with latency d between a pair of sequences X and $Y$is therefore given as:

$$r(d) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_{i-d} - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_{i-d} - \bar{y})^2}} \quad (1)$$

where $\bar{x}$ and $\bar{y}$ are the sample means of X and Y.

B. Dynamic Time Warping

Dynamic Time Warping (DTW) is a pattern recognition technique used for obtaining a measure of similarity between two sets of sequentially sampled points. DTW accomplishes this by computing the deformation cost accrued when aligning one set of points to the other. The goal of DTW is to solve

the correspondence problem between the pair of sequential points by employing an efficient elastic analysis to compute an accurate and informative similarity metric. DTW alignment cost can be treated as inverse similarity or inverse-synchrony.

C. Riemannian elastic metric-based curve alignment [25]

The basic idea of this elastic metric is that it gives the cost required to connect the points with arbitrary paths and to iteratively straighten the paths, using the gradient of an energy function, until the path becomes a geodesic. This framework is general enough to be applied to closed curves in $\mathbb{R}^2$. Our goal is to investigate how well this metric captures the changes that occur between interlocutors, during conversation.

If we treat the feature-under-investigation for each interlocutor (action units, pose points, *etc.*) as functions on an interval $[a, b]$, then, given two curves f_1 and f_2, we need to find a warping process $\gamma : [a, b] \rightarrow [a, b]$, so that f_1 is optimally aligned to $f_2 \circ \gamma$. The resulting alignment criteria should result in proper distances between aligned functions such that the solutions are symmetric *i.e.* the optimal alignment of f_1 to f_2 is the same as that of f_2 to f_1. Srivastava *et al.* [25] presented a distance formulation, the square-root velocity function (SRVF) which combines the strengths of an elastic metric with a path straightening method for finding geodesics.

The SRVF of a function f is given as:

$$q(t) = \text{sign}(\dot{f}(t))\sqrt{|\dot{f}(t)|} \tag{2}$$

Hence, if q_1 and q_2 are the SRVF for f_1 and f_2, and γ is the warping coefficient, then,

$$\|q_1(t) - q_2(t)\| = \|q_1 - (q_2(t) \circ \gamma)\sqrt{\dot{\gamma}}\| \tag{3}$$

The goal now is to solve for the optimal $\gamma*$ that preserves distances under identical mappings (isometric property). For interactional synchrony measures, visual cues of synchrony are obtained as discrete time points so dynamic programming is used to solve for optimal alignment.

In a nutshell, the algorithm uses the visual cue values from each of the three interlocutors to form a finite grid over $[0, 1]^3$. It then seeks to compute the optimal piecewise linear warping function passing through the grid points.

This metric is similar to DTW rather than working with Euclidean distances in the space of the original curve functionals, the elastic metric performs the alignment using geodesics. The input to the algorithm consists of the two curves being evaluated; when curves are of different lengths, they are re-sampled to have the same number of points. The q function is applied to two curves and the optimal re-parameterization is applied from one curve to the other using dynamic programming. The geodesic distance between the registered curves is then computed as the final alignment or warping cost, which we again interpret as inverse-synchrony.

V. Evaluations Based on Curve Alignment

A. Computing synchrony metrics

We compute the correlation coefficients, DTW costs and elastic metrics, first using all the AU values and then using the three most intense AUs across each group as described previously. Similarly, we compute the correlation coefficients and DTW costs separately on the coordinates of the seven upper body keypoints. The following describe the different measurements computed:

1) **AllAUCorr**: For each video, we consider all 18 facial AU across 1800 frames and three user pairs, and calculate the correlation coefficient between each pair of individuals, i.e.- Individual A-B, Individual B-C, Individual A-C, in each video for all the 18 AUs. We then sum the correlation coefficients of the 3 pairs of individuals to get one aggregate value for each group. We do this for each of the 32 groups.
2) **AllAUDTW**: We perform exactly the same process as described above but instead compute the DTW alignment cost. Since DTW is a cost measure, we take the inverse of the three DTW values and sum across these inverted cost values to get one value for each group. We do this for each of the 32 groups.
3) **3AUCorr**: For each video, we consider all 18 facial AUs across 1800 frames but use only the 3 AUs with the highest values across individuals. Similar to calculating *AllAUCorr* above, we compute this feature using only three action units.
4) **3AUDTW**: We perform exactly the same process as described above but instead compute the DTW alignment cost using only the top 3 AUs.
5) **3AUElastic**: We perform exactly the same process as described above for DTW but instead compute the elastic metric, which is also a form of alignment cost. Similar to the process above, we use only the top 3 AUs. Since the elastic metric is a cost measure, we take the inverse of the three elastic metric values and sum across these inverted costs to get one value for each group. We do this for each of the 32 groups.
6) **Pose7DTW**: We perform exactly the same process as described above but instead consider the 7 upper body joints across a large number of chosen frames and calculate the DTW alignment costs for each video.
7) **Pose7Corr**: For each video, we consider the 7 upper body joints across a large number of chosen frames and calculate the correlation coefficient between each pair of individuals, i.e.- Individual A-B, Individual B-C, Individual A-C, in each video for all the 7 keypoints. We then sum the correlation coefficients of the 3 pairs of individuals to get one aggregate value for each group. We do this for each of the 32 groups.

B. Comparisons with ground-truth

In order to compare the the different metrics calculated with the annotated label values, we again run Pearson's correlation between the human-labeled values and the metrics computed by each alignment technique. Table I shows the resulting correlation values, indicating how well the metrics match up with the annotated label values.

TABLE I
CORRELATIONS WITH HUMAN-ANNOTATED SYNCHRONY SCORES

Feature combination	Corr with GT
AllAUCorr	**0.3833**
AllAUDTW	0.0318
3AUCorr	0.3667
3AUDTW	0.0736
3AUElastic	0.0318
Pose7DTW	0.0264
Pose7Corr	0.0252

C. Discussion on curve alignments

From the comparisons provided in the work, we observe that although correlation is a relatively simple matching method, it gives the closest to human-perception synchrony measures, when compared with more sophisticated curve matching techniques such as DTW and elastic curve alignment metric tested on both the facial and body parts/pose features. When compared with the human generated label values, the correlation coefficient resulted a correlation value of 0.3833 and 0.3667 for facial features; while the correlation value was 0.2634 and 0.2052 between the upper body pose features and the human annotations. Both are statistically significant, when compared with the other measures and combinations of features. Correlation measures of facial and/or body features can therefore be used as a quantifiable and repeatable metric for measuring interactional synchrony among two interlocutors.

VI. DEEP NETWORK FOR PREDICTING SYNCHRONY

Because the facial features were shown to consistently outperform body movements, as observed from the experiments in Section IV, we train various regression neural networks on the facial features to predict the extent of synchrony between the interlocutors. We train our main proposed end-to-end LSTM network using a regression loss function and follow with training other supporting regression networks, to investigate their efficacy in estimating synchrony. We involve the supporting methods in order to determine if the use of coupled features (features extracted from 2 signals simultaneously) can yield better results than the end-to-end LSTM, when estimating interactional synchrony. To this end, the main proposed method is trained all at once, while the other supporting methods involve a 2-step process where (i) time-based features are extracted from coupled data; and (ii) those features are fed into a neural network with a regression loss function.

All the methods used the same regression architecture on the same GFT data, where each video is one-minute or 1800 frames long. We used an overlapping window of size 30 and fed the sub-signals to various architectures. We split the entire dataset consisting of 32 groups into 25 for training and the remaining 7 for testing, and rotate through for cross-validation.

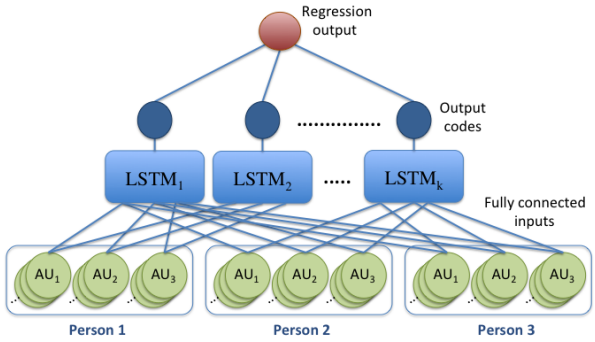

Fig. 3. An LSTM network to predict the synchrony among 3 sets of input signals, where each set represents a subject. In this diagram, each set has 3 channels - the three strongest AUs. For the sake of clarity, not all the connections are shown in the input layer, but in reality, every channel of data from each subject is connected to all the input nodes of every LSTM network. Note: the number of LSTM networks does not necessarily equal the number of subjects at the input.

A. Method 1: The end-to-end LSTM network

The main proposed model is an end-to-end LSTM network, shown in Figure 3, where all the action unit sequences for all the three subjects are simultaneously fed into every LSTM in the network. The inputs consisted of 3 sets of 3 channels each (the 3 most prominently changing AUs). In the input layer, each LSTM network has a neuron for each channel of the input for all three individuals.

A total of six LSTM nodes were used to learn coupling in our dataset. The LSTM network output feeds directly into a neural network regression module, and learning is performed end-to-end. We employ this architecture to learn the extent of interaction between two or more sets of input signals. The network utilizes a lookback feature of one second (or 30 frames). It is important to note that the number of LSTM networks does not necessarily have to be equal to the number of subjects or channels at the input. More information is provided on how we selected the optimal number of input networks in Section VI-A1. All neurons use the ReLU activation function.

During training, the sub-signals were assigned labels by human annotators who assigned the value to the entire group of three, as a measure of their perceived level of interaction - between 1 and 5. A simplified model can be seen in Figure 3.

1) Estimating the number of LSTMs required: In the testing methodology, the time series data is input to the LSTM architecture and the output codes from all the LSTMs are trained using a regression-based mean-squared error (MSE) loss (Figure 5). To determine the optimal number of LSTM nodes required, we fixed a working version of the architecture and recorded the losses when varying the number of input LSTM networks, from 1 to 9 as shown in Figure 4. The optimal number of LSTMs for this dataset was 6, although it

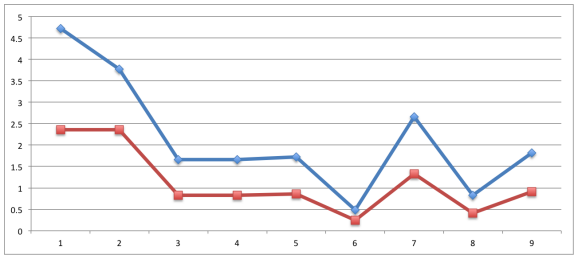

Fig. 4. x-axis is the number of LSTM networks and y-axis is the lowest error obtained from the model on the same data. The red (or lower curve) represents the results obtained from training data and the blue (or upper) represents the results obtained from the validation data.

can be seen that the overall number of LSTMs in this range, makes only a small difference to the end resu

B. Method 2: The two-step regression networ features

Coupled features are extracted on pairs o ments, where each AU segment was only 30 fr coupled features we were interested in included from autoencoder pairs, those extracted using coefficients, DTW and elastic metrics. For al was important to separate the training from the group level rather than at the segment l segments from a group are used in training, from the same group cannot be used in testin

1) Features from an LSTM autoencoder: A pling autoencoder is trained to predict the nex signals when presented with an input set of s same sequence. The purpose here is to lea parameters that encode any interactions bet signals, as well as the parameters that gover at time $t+1$ given the value of time t, for ea

The LSTM is trained until convergence on t and for feature extraction, new data is input for the representative code generated is the featur

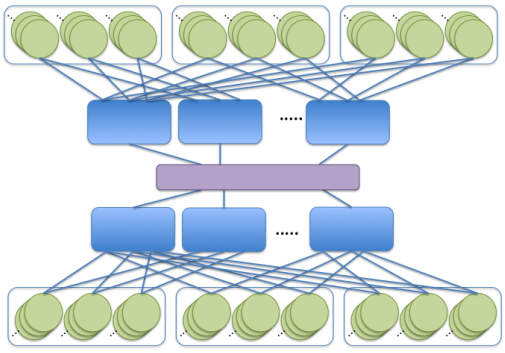

Fig. 5. An LSTM based autoencoder used to extract time-based codes with any coupling information embedded within the codes (the purple bar). Image best viewed in color.

The resulting codes were then fed into a traditional neural network with the mean-squared-error loss function, for predicting real-valued human annotations of perceived synchrony.

2) Features from curve alignment techniques: For the data setup here, segments of AU data were extracted from the videos and correlation coefficient and DTW curve alignment techniques were applied to each segment pair. Unlike the autoencoder which simultaneous encodes all three participants, the curve alignments compute scores on pairs of segments and all three combinations of scores are fed into regression-based neural network as shown in Figure 6. Each combination of segment pairs inherits the synchrony label of the entire 3-person video. For example, for a triadic video involving participants ABC, we break the video into overlapping seg-

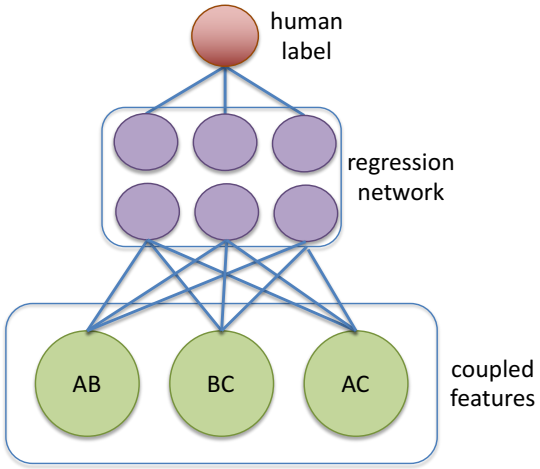

Fig. 6. The regression network which takes the coupled features from the combinations of pairs of participants as inputs, and the human labeled synchrony values as targets. Image best viewed in color.

C. Results and discussion on predicting synchrony

To test the effectiveness of our model, we computed the mean of the absolute percent error μ_e given as: $\mu_e = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right|$

where N is the number of samples tested, Y_i is the target value and $\hat{Y}_i$ is the final value predicted by the regression network. We compute this for the end-to-end network.

The full set of results on the real-life dataset are shown in Table II below.

We observe that our proposed end-to-end model can learn a measure of synchrony in-step with the human annotated labels. It performs better than the other two-step techniques tested against it, including those based on features extracted from the LSTM auto-encoder. This is because the proposed model trains end-to-end from feature extraction to target matching and propagates its losses over the entire network. This is

Method	Mean Abs Error (Training)	Mean Abs Error (Testing)
LSTM end-to-end	0.1728	0.3551
Autoencoder features	0.4551	0.5492
Correlation features	0.6588	0.7235
DTW features	0.6609	0.65315

TABLE II
BEST RESULTS AFTER RUNNING 5-FOLD VALIDATION ON THE DATASET

unlike the other models where any errors incurred in the feature extraction stage is simply transferred to the regression stage and there is no opportunity to propagate such errors. From comparing the training and testing errors, it is not even clear that the regression network is learning any patterns when provided the input features extracted from the two curve alignment techniques tested.

Furthermore, although the LSTM based end-to-end architecture has the modeling power of traditional deep networks, it also works well in the presence of limited data as is the case with the GFT dataset. Hence, based on the analysis and results reported, we can be confident that the proposed architecture looks to measure the extent of synchrony even in presence of limited data.

VII. CONCLUSION

We have successfully addressed the main research questions we intended to address at the onset of this study. We extensively analyzed the use of different curve alignment techniques to determine whether human perceived interactional synchrony is better modeled via facial expressions or body pose movements. We observed that facial expressions consistently outperformed body movements, and the standard correlation coefficient estimation was the best-in-class time series matching technique (from the tests we performed).

We have also shown that in the presence of limited data, we are able to computationally detect and measure the extent of interactional synchrony with minimal errors, when trained with metrics representing human perceptions of synchrony. We compared several classical methods for evaluating time-series data with our proposed approach and show that they do not perform as well as an end-to-end recurrent neural network in the presence of multiple interacting signals.

REFERENCES

[1] K. Hogan and R. Stubbs, *Can't Get Through: 8 Barriers to Communication*. Grenta, LA: Pelican Publishing Company, 2003.

[2] F. J. Bernieri and R. Rosenthal, *Fundamentals of Nonverbal Behavior*, ser. Interpersonal coordination: behavior matching and interactional synchrony. Cambridge University Press, 1991.

[3] T. L. Chartrand and J. A. Bargh, "The chameleon effect: the perception–behavior link and social interaction," *J. Pers. Soc. Psychol*, vol. 76, no. 2, pp. 893–910, 1999.

[4] L. Yu and M. Tomonaga, "Interactional synchrony in chimpanzees: Examination through a finger-tapping experiment," *Scientific Reports*, no. 5, 2015.

[5] A. Edmondson, "Psychological safety and learning behavior in work teams," *Administrative Science Quarterly*, vol. 44, pp. 350–383, 1999.

[6] F. Ramseyer and W. Tschacher, "Nonverbal synchrony in psychotherapy: Coordinated body movement reflects relationship quality and outcome," *J. of Cons. and Clin. Psychol.*, vol. 79, pp. 284–295, 2011.

[7] T. L. Chartrand and J. A. Bargh, "The chameleon effect: The perception-behavior link and social interaction." *Journal of Personality and Social Psychology*, vol. 76, no. 6, p. 893–910, 1999.

[8] E. Delaherche, M. Chetouani, A. Mahdhaoui, C. Saint-Georges, S. Viaux, and D. Cohen, "Interpersonal synchrony: A survey of evaluation methods across disciplines," *IEEE Transactions on Affective Computing*, vol. 3, no. 3, pp. 349–365, July 2012.

[9] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden markov models for complex action recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, ser. CVPR, 1997.

[10] I. Rezek, P. Sykacek, and S. J. Roberts, "Learning interaction dynamics with coupled hidden markov models," *IEE Proceedings - Science, Measurement and Technology*, vol. 147, no. 6, pp. 345–350, Nov 2000.

[11] A. W. Woolley, C. F. Chabris, A. Pentland, N. Hashmi, and T. W. Malone, "Evidence for a collective intelligence factor in the performance of human groups," *science*, vol. 330, no. 6004, pp. 686–688, 2010.

[12] R. Li, J. Curhan, and M. E. Hoque, "Predicting video-conferencing conversation outcomes based on modeling facial expression synchronization," in *IEEE International Conference on Automatic Face and Gesture Recognition*, ser. FG, 2015.

[13] X. Yu, S. Zhang, Y. Yu, N. E. Dunbar, M. L. Jensen, J. K. Burgoon, and D. N. Metaxas, "Automated analysis of interactional synchrony using robust facial tracking and expression recognition," in *IEEE International Conference on Automatic Face and Gesture Recognition*, ser. FG, 2013.

[14] Z. Hammal, T. E. Bailie, J. F. Cohn, D. T. George, J. Saraghi, J. N.Chiquero, , and S. Lucey, "Temporal coordination of head motion in couples with history of interpersonal violence," in *IEEE International Conference on Automatic Face and Gesture Recognition*, ser. FG, 2013.

[15] W.-S. Chu, J. Zeng, F. De la Torre, J. F. Cohn, and D. Messinger, "Unsupervised synchrony discovery in human interaction," in *International Conference on Computer Vision*, ser. ICCV. CVF/IEEE, 2015.

[16] H. Hung and D. Gatica-Perez, "Estimating cohesion in small groups using audio-visual nonverbal behavior," *IEEE Transactions on Multimedia*, vol. 12, no. 6, p. 563–575, 2010.

[17] S. Bilakhia, S. Petridis, A. Nijholt, and M. Pantic, "The MAHNOB Mimicry Database-a database of naturalistic human interactions," *Pattern Recognition Letters*, vol. 3, 2015.

[18] J. R. Terven, B. Raducanu, M. Meza, and J. Salas, "Evaluating real-time mirroring of head gestures using smart glasses," in *IEEE International Conference on Computer Vision Workshop*, ser. ICCVW.

[19] J. M. Girard, W.-S. Chu, L. A. Jeni, J. F. Cohn, and F. De la Torre, "Sayette group formation task (gft) spontaneous facial expression database," in *IEEE International Conference Automatic Face and Gesture Recognition (FG)*, ser. FG, 2017.

[20] M. A. Sayette, K. G. Creswell, J. D. Dimoff, C. E. Fairbairn, J. F. Cohn, B. W. Heckman, T. R. Kirchner, J. M. Levine, , and R. L. Moreland, "Alcohol and group formation: A multimodal investigation of the effects of alcohol on emotion and social bonding," *Psychological Science*, vol. 23, no. 8, pp. 869–878, 2012.

[21] T. Baltrusaitis, P. Robinson, and L. P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *2013 IEEE International Conference on Computer Vision Workshops*, Dec 2013, pp. 354–361.

[22] P. Ekman and E. L. Rosenberg, *What the Face RevealsBasic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*. New York, NY, US: Oxford University Press, 2005.

[23] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, in *IEEE Conference on Computer Vision and Pattern Recognition*.

[24] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Real-time multi-person 2d pose estimation using part affinity fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR, 2017.

[25] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn, "Shape analysis of elastic curves in euclidean spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 7, pp. 1415–1428, Jul. 2011.