Automated Analysis of Interactional Synchrony using Robust Facial Tracking and Expression Recognition

[1]Xiang Yu, [1]Shaoting Zhang, [1]Yang Yu, [2]Norah Dunbar,[2]Matthew Jensen,
[3]Judee K. Burgoon, [1]Dimitris N. Metaxas

[1]Center for Biomedical and
Image Modeling
Rutgers University
617 Bowser Road, Piscataway,
N.J, USA
{xiangyu,shaoting,yyu,dnm}@cs.rutgers.edu

[2]Center for Applied Social
Research
University of Oklahoma
610 Elm Avenue Norman,
O.K, USA
{ndunbar,mjensen}@ou.edu

[3]Center for Management of
Information
University of Arizona
1130 East Helen Street Tucson,
A.Z, USA
jburgoon@cmi.arizona.edu

***Abstract*— In this paper, we propose an automated, data-driven and unobtrusive framework to analyze interactional synchrony. We use this information to determine whether interpersonal synchrony can be an indicator of deceit. Our framework includes a robust facial tracking module, an effective expression recognition method, synchrony feature extraction and feature selection methods. These synchrony features are used to learn classification models for the deception recognition. To evaluate our proposed framework, we have conducted extensive experiments on a database of 242 video samples. We validate the performance of each technical module in our framework, and also show that these synchrony features are very effective at detecting deception.**

I. Introduction

Detecting deception in interpersonal dialogue is challenging since deceivers take advantage of the give-and-take of interaction to adapt to any sign of skepticism in the interviewer's verbal and nonverbal feedback [3], [25]. That same give-and-take of interaction, however, has the potential to offer subtle clues to deception through the disruption of interactional synchrony. Interactional synchrony refers to interaction that is non-random, patterned, and aligned in both timing and form. It is considered a key marker of interaction involvement, rapport, and mutuality. The simultaneous synchrony or speaker-listener synchrony means that two or more people's behaviors mimic or match one another (e.g., similar postures and facial expressions) in the same time frame and behavioral changes occur at the same junctures. In addition, concatenous synchrony occurs when one partner's behavior follows the other rather than occurring simultaneously [7]. In either case, engaging in deception may disrupt interactional synchrony and may therefore be a clue to the presence of deceit. Practitioners have suggested using rapport-building techniques or interactional synchrony as an effective method for detecting deception in a variety of law enforcement settings [22].

Synchrony has been widely used in analyzing non-verbal human actions[20]. Prabhakar et al.[18] prompted the temporal causality model for deducting visual events. Morency et al.[17] proposed a recognition-based method to investigate the human interactions. Non-verbal actions always exhibit along time axis. Thus temporal causality is a natural clue to analyze the property of the visual events. Traditionally, to differentiate deception from truth, people adopt visual or audio methods[1], [24]. In those methods, which are unaided by technology, the average performance is estimated at 54%, or slightly above chance[2]. With the technology development of computer vision, using automated visual strategies to help analyze synchrony and further recognize deceiving has become another mainstream of deception detection. There are automatic extraction ways of the deceptive behavior cues[13]. Natural language processing is used to discriminate deceptive speech[10]. In combining, multi-modality strategies (manual coding with automated analysis[30], automated visual and audio analysis[15], etc) are fusion methods expected to improve the detection accuracy.

Video taping is one of the multi-modality ways to help analyze synchrony, in which there are both visual actions and speech content. The dataset of videos for this paper is constituted from a cheating experiment in which some subjects cheated during a trivial dialogue and some did not, but all of them were encouraged to appear as credible as possible. Among the videos, some of the participants were interviewed face to face and others were interviewed with computer mediated communication via Skype. Part of the cheaters confessed their cheating behavior during or after the dialogue interview. Such interviews were raised by certified professional examiners supplied by a federal agency and the dialogue includes three phases of questioning: (1) a set of baseline questions that were benign; (2) a set of questions that presented indirect accusations; (3) a final set which directly inquired about cheating. Each dialogue is conducted by one such examiner and one participant, in which the participant may appear to be deceiving.

In this paper, we propose an automated framework to represent interpersonal synchrony, and to investigate whether such interpersonal synchrony can be an indicator of deceit. The framework starts with a face tracking module. With key positions provided by the face tracker, i.e. eye points, mouth

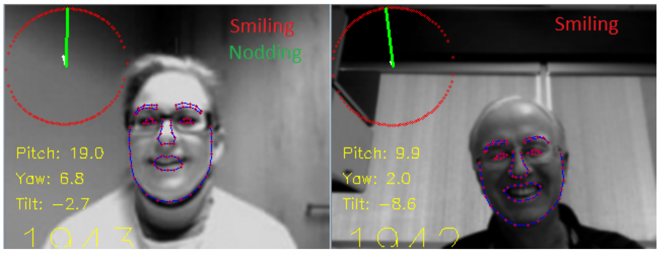


Fig. 1. Sample snapshots from tracked facial data showing a subject (left) and an interviewer (right). Red dots represent tracked facial landmarks (eyes, eyebrows, etc.), while ellipse in top left corner depicts the estimated 3D head pose of the subject; top right corners show the detected expressions and head gestures for subject and interviewer.

points, face profile points, etc, we designed an expression detection module and head pose gesture module to extract the deceptive behavior cues. Based on such cues, we developed cross correlation based method to extract synchrony feature. The synchrony feature is finally sent to our specifically designed learning based classifier to discriminate deception. We have conducted extensive experiments to validate the performance of using synchrony features. Experimental results show that this type of feature is very effective at detecting deception.

II. Methodology

Our main contribution in this work is designing a technical framework to depict interactional synchrony and further to discriminate truth from deception. The technical framework includes tracking facial movements module, expression recognition module and head pose detection module, as shown in Figure 1. Based on the lower level features extracted by those modules, i.e., head nodding, head shaking, smile, etc, we designed the temporal causality like strategy to generate higher level synchrony feature. Using the higher level feature, a data-dependent learning based classifier is designed to differentiate deceptive groups from truthful groups. The whole flowchart is shown in Figure 2. Each of the modules is illustrated in detail in the following subsections.

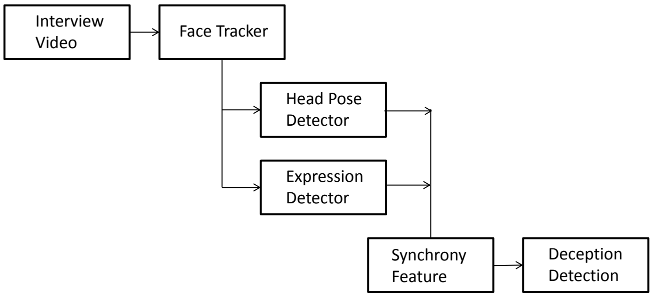
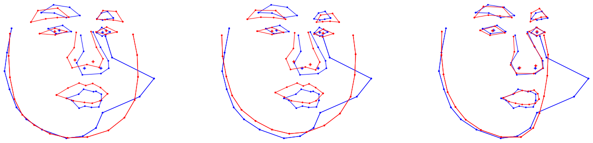


Fig. 2. The workflow of deception detection framework consisting face tracker, head pose detector, expression detector and synchrony feature extractor and deception detection classifier.



(a) E-ASM [16] (b) SSR [27] (c) Ours

Fig. 3. We fit a shape to the observations (blue contour). The observations contain both Gaussian noise and sparse outliers. In addition, they have been applied a large rotation to simulate multi-pose. From left to right, red contours are the results using E-ASM [16], sparse shape registration [27] and our method, respectively.

A. Sparse Shape Representation based Multi-pose Face Tracking

In natural scenes, i.e. dialogues, interviews, etc, people do not always keep their face frontal and sometimes part of their faces are occluded by objects or themselves. Furthermore, poor lighting conditions may cause a low contrast image or cast shadows on faces, which make the face tracking problem challenging.

We have developed a robust face tracker [12] based on the Active Shape Models(ASMs) [6]. ASM achieves significant success on facial feature localization, and many variations have been proposed [11], [19], [16]. Due to the computational expensiveness in fitting ASMs in every frame, we track the features using the KLT tracker [21] across consecutive frames. The KLT tracker is a method for registering two local features and computes the displacement of the feature by minimizing the intensity matching cost.

As there are pose variation and occlusion problems mentioned above, inspired by and closely related to the sparsity techniques [5], especially the sparse representation [26] and sparse shape models [27], [31], [32], we use sparse representation to address those problems in our ASM-type framework.

Notations: Given a shape model containing n landmarks, we define the shape as a vector $\mathbf{s}$, which is formed by concatenating the x and y coordinates of all the landmarks. $\mathbf{s} = [x_1, y_1, x_2, y_2, \cdots, x_n, y_n]^T$. The shape prior constraint in the ASMs assumes that the shapes follow a Gaussian distribution in a low dimensional subspace. The distribution is estimated based on the training sample shapes, which are represented as a $2n \times m$ matrix S for m samples. The mean shape $\bar{\mathbf{s}}$ is the average of all training samples, and the covariance matrix Σ is approximated by the first k principle components as $U \Lambda U^T$, where Λ is a $k \times k$ diagonal matrix containing the largest k eigenvalues and U is a $2n \times k$ matrix containing the corresponding k eigenvectors. A new shape $\mathbf{s}$ in the k dimensional subspace is represented as: $\mathbf{s} = \bar{\mathbf{s}} + U\mathbf{b}$, where $\mathbf{b}$ is a vector with length k for the coefficients. In the shape registration problem, assuming there is an estimated shape $\hat{\mathbf{s}}$ from local detectors, we expect to find a shape $\mathbf{s}$ that is not only similar to $\hat{\mathbf{s}}$, but also follows the shape distribution from the training samples. This can be achieved by minimizing the following energy function

$$\arg\min_{\mathbf{b}} \mathbf{b}^T \Lambda^{-1} \mathbf{b} + \|\hat{\mathbf{s}} - (\bar{\mathbf{s}} + U\mathbf{b})\|_2^2 \qquad (1)$$

where the first term is the Mahalanobis distance, and the second is the distance between $\mathbf{s}$ and $\hat{\mathbf{s}}$.

Handling Occlusions: This setting works well when the errors of landmark detections are not large. However, the detection results may be far away from the correct positions if some of the face are occluded. The models following Gaussian distribution are sensitive to these gross errors, so additional term should be introduced to model these errors. Based on our observation, the large errors from partial occlusion are often sparse compared to the whole data. Therefore, they are explicitly modeled as a sparse vector $\mathbf{e} \in \mathbb{R}^{2n}$:

$$\arg\min_{\mathbf{b},\mathbf{e}} \mathbf{b}^T \Lambda^{-1} \mathbf{b} + \|\hat{\mathbf{s}} - (\bar{\mathbf{s}} + \mathbf{U}\mathbf{b} + \mathbf{e})\|_2^2$$
$$\text{s.t. } \|\mathbf{e}\|_0 \leq k_1 \qquad (2)$$

where $\|\cdot\|_0$ is the L_0 norm, which is the number of nonzero elements, and k_1 is the sparse number of $\mathbf{e}$. The nonzero elements of $\mathbf{e}$ capture the sparse large error, while L_2 norm loss function can deal with small errors. Similar formulation has been proposed in [27], and shows promising performance for front facial feature localization with partial occlusions.

Handling Multi-Pose: This model achieves good performance when the pose of the head is unchanged. If there is a large change of the head pose, the projected 2D shapes will be dramatically different from the mean, and are not limited to a low dimensional subspace. One solution is to model them as a linear combination of training shapes containing different poses. Since the training shapes are usually overcomplete and may contain noise, the linear representation should also have a constraint on the number of non-zero elements:

$$\arg\min_{\mathbf{w},\mathbf{e},\mathbf{t}} \|\hat{\mathbf{s}} - (S\mathbf{w} + \mathbf{e})\|_2^2$$
$$\text{s.t. } \|\mathbf{e}\|_0 \leq k_1, \|\mathbf{w}\|_0 \leq k_2 \qquad (3)$$

where $\mathbf{w}$ is the weight for each training sample, and k_2 is the sparse number of $\mathbf{w}$. Different from most previous methods handling multi-poses, the sparse linear representation does not require face pose estimation. The faces under different poses are registered under the same framework. Meanwhile, the facial shapes with similar pose are more likely to have larger weights, which implicitly show the face pose.

Fig. 3 shows the comparison of ASM, shape registration (i.e., modeling the sparse outliers), and our method. The results show that our method can handle both sparse outliers and pose variations.

B. Gesture and Facial Expression Detection

From the landmark positions in each frame, we are able to estimate the 3D poses (pitch, yaw and tilt) and detect the relevant gestures (head shaking and nodding). To estimate the face pose, we built a linear regression model for each linear region in the shape manifold. The regression model takes the X and Y coordinates of the 79 landmarks as input, and predict the pitch, yaw and tilt angles.

The face nodding is rapidly and repeatedly moving the face up and down. By differentiating the pitch value in each frame, we are able to detect the face nodding. As shown in Figure 4, the top row shows a sequence of face shapes; the middle row shows the corresponding 3D head poses; and the bottom row shows the plot of head pitch angle against time, in a characteristic pattern of head nodding. We use a similar approach to detect the head shaking by differentiating the yaw angles.

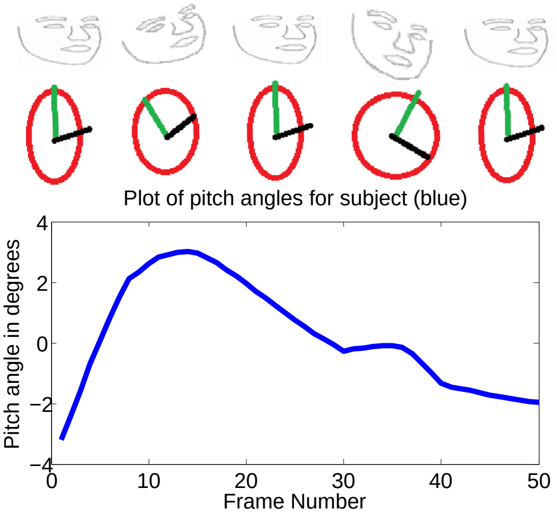


Fig. 4. Detecting head nodding. **Top**: face shapes. **Middle**: face poses. **Bottom**: The change of head pitch angles against time.

Also we have built a facial expression classifier to detect facial expressions such as smiles [29], [28]. We use a ranking-based framework of facial expression analysis, which can recognize expressions and estimate its intensity. Our method is based on an observation that the pair-wise ordinal relationship along the temporal domain is obvious, despite the short of quantitative measurement of intensity. Therefore, it is relatively easy to model the intensity estimation as a ranking problem. Our framework consists of three components: 1) facial appearance feature representation. We use the haar-like features to represent facial appearance due to its good properties, especially in facial appearance representation [23]; 2) Ordinal pair-wise data organization. It describes how to organize the data to make it suitable for the ranking model; 3) Building ranking model. This is the core component. Due to a large number of haar-like features existing, we propose to use the RankBoost [9] to select a subset of haar-like features to build a final strong ranker. In order to further improve the performance, we introduce the l1 based regularization into the RankBoost. The final ranking score given by a ranking function can be used for expression intensity estimation and recognition.

C. Synchrony Features

The subtle and significant way people influence each other can be seen through their nonverbal synchrony. Synchrony refers to similarity in rhythmic qualities and enmeshing or coordination of the behavioral patterns of both parties in an interaction [4]. Such synchrony can either be simultaneous or concatenous. In Dunbar et al.'s work [7], synchrony is found in a variety of nonverbal cues including gestures, nodding or shaking, facial mirroring, etc. When providing pairs of interview videos, we could obtain head nodding or shaking and facial expressions (especially smiling) in the videos by our proposed facial tracking and facial expression detection methods. Based on such lower level features, we intend to check the simultaneous or concatenous responses from both two people in one interview.

Lower level feature vectors of two interview videos from one interviewer and one interviewee can be viewed as two corresponding data sequences. We know that we could get large responses while doing correlation over two sequences if the two sequences has similar magnitude at the same position, which could measure the simultaneous response. If two sequences has similar magnitude at different positions, we could take a time sliding window to compensate the time delay and then calculate their correlation. Cross correlation is a standard method of estimating the degree to which two sequences are correlated. The definition of cross correlation of two signal series of which one delayed at gap d is as:

$$C(d) = \frac{\sum_i (x_i - \bar{x})(y_{i-d} - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2}\sqrt{\sum_i (y_{i-d} - \bar{y})^2}} \tag{4}$$

where x_i and y_i are the ith element of sequence x and sequence y, $\bar{x}$ and $\bar{y}$ are the mean value of sequence x and sequence y.

In order to accommodate with concatenous synchrony, we divide two sequences into overlapped time slots. Firstly the two sequences are required to have the same length. Then we equally divide each sequence into m time slots. Starting from either of the sequences, for current time slot, we will go backward of t time slots and go forward of t time slots to calculate their correlation with the current time slot. And we choose the largest cross correlation value as the current time slot's feature value. We repeat such procedure for every time slot in one sequence until the end of the sequence. Thus we could obtain a cross correlation based higher level feature vector with length m.

III. Experiments

In this section, we built up a database of 242 video samples. Those video samples are conducted by expert interviewers and student interviewees through computer mediated communication. This experiment follows the rule illustrated in [8]. Half of them are videos of interviewers. The rest are the corresponding interviewees' videos. Thus we have 121 pairs of interviews for further analysis. Due to data validity consideration, we selected 100 out of 121 pairs of videos as our training and testing data. These video pairs vary from 4500 frames to 15000 frames. Although video pairs' lengths are different, we ensured each video pair keep the same length (since the start of the video pair is the same, we cut the longer one complying with the shorter one if there is length difference), which allowed using a fixed number of time slots to analyze the video sequences.

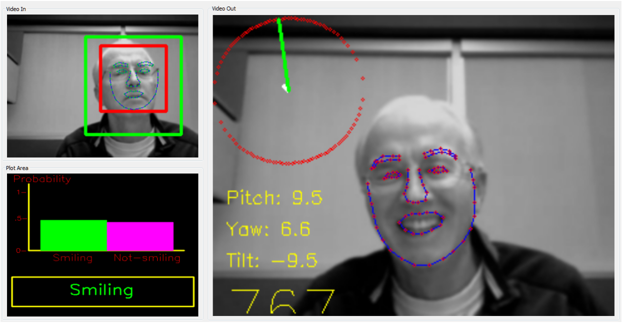


Fig. 5. Face tracking and expression, head pose estimation visual result. **Left Top**: Initial face detection and landmark initialization. **Left Bottom**: Score plot of expression (smile or not smile) recognition. **Right**: Facial landmark tracking result and head pose estimation(depicted by pitch, yaw and tilt).

In the synchrony detection step, we have extracted head nodding, head shaking, smiling and looking forward facial gestures. The visual result is shown in Figure 5. Head motion can be detected by analyzing pitch, yaw and tilt as demonstrated in section 2.3. Further we combine the lower level feature vectors to form higher level features. As our claimed dyadic strategy, we believe that efficient communication must have response from each other during the conversation. Thus correlation based method is adopt to analyze synchrony. Our classification task is to tell the truthful group from the deceptive group. During the feature selection part, each step we separately train a feature selector by Genetic Algorithm.

A. Evaluation of Features

Before sending features into classifiers, the different types of features should be investigated, of which it is effective for classification. Our major strategy is to leave each single feature out of the whole feature vector and then test the recognition accuracy. Also, we will give out the single feature's recognition accuracy and visualize the feature vector in plots to see the distinguishability of each feature. Basically we have 4 types of features, head nodding feature, head shaking feature, smiling or not smiling feature and look forward or look away feature. During this step, we will look into the average precision of 2 class (truthful and deceptive) classification to evaluate each feature vector. The accuracy in this evaluation and the following evaluations is derived by repeated trials with same experimental setups. Table I shows the average precision of different feature combinations over 2 class classification.

From Table I, we could see that when feature "Nod" or feature "Shake" is excluded from the whole feature vector,

TABLE I

EVALUATION OF FOUR FEATURES, I.E., “NOD”, “SHAKE”, “SMILE” AND “LOOK FORWARD”. EACH ROW MEANS ONE SPECIFIC FEATURE. “ALL-BUT-ONE” MEANS THAT ALL FEATURES ARE USED EXCEPT THE ONE IN THAT ROW.

Features	**All-but-one**	**Single feature**
Nod	0.641(±0.076)	0.64(±0.035)
Shake	0.689(±0.069)	0.68(±0.021)
Smile	0.644(±0.068)	0.60(±0.063)
Look forward	0.607(±0.074)	0.61(±0.048)

the performance would be higher than the rest ones. And when feature “Smile” or “Look forward” is excluded, the performance would drop. When testing each single feature’s accuracy, it appears the trends that “Nod” and “Shake” are more significant than “Smile” and “Look forward”. This may because “Node” or “Shake” is instantaneous action while “Smile” and “Look forward” last for a while. In evaluating the synchrony features, instantaneous features reflect synchrony more direct than the last-for-a-while features.

In Figure 6, the vertical dot lines separate the plot into 4 regions. The first column indicates feature “Nod”, the second one is feature “Shake”, the third one is feature “Smile” and the last one is feature “Look forward”. We plot the average feature vector of each group in the subplots. With black line drawn in the figure, we could see that in region three, the pattern of feature vector is obviously different, in subplot of truthful, it is going down; in subplot of deceptive, it is going up. While in region four, the average value of those numbers is going down from above 0.9 until around 0.8. The average feature vector difference shows us that the synchrony features for truthful group and deceptive group are obviously different.

B. Evaluation of Feature Selection

Once we obtain effective lower level features, a higher level feature is formed combining those lower level features. But such higher level feature is not good enough for the classification task, which can be reflected by Table I, the 2 class classification average precision is just over 60%, while the baseline accuracy is provided as 54%[2]. Due to the lower level feature extraction, noise may be brought in. Nevertheless, inside the higher level feature, different feature elements may have correlation with each other. Thus we intend to choose the feature elements out of the original higher level feature vector in order to remove noise and redundancies. In this work Genetic Algorithm (GA) is adopted to select features.

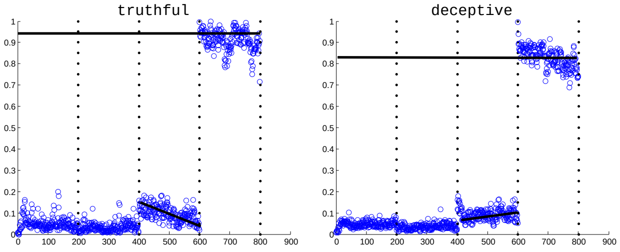


Fig. 6. Mean feature vector patterns of two groups. **Left:** truthful group’s mean feature vector pattern. **Right:** cheating group’s mean feature vector pattern.

TABLE II

EVALUATION OF USING FEATURE SELECTION.

	No Selection	Feature Selection
Accuracy	0.644(±0.040)	0.728(±0.053)

We list the accuracy comparison of features with and without feature selection in Table II. The accuracy with feature selection is about 10% higher than that without feature selection, which indicates that the feature selection is a key step of accuracy improvement.

C. Evaluation of the Two-Class Classification

Even with feature selection’s promotion, it is still possible to improve the accuracy because proper classifier design could enhance performance too. It is a 2 classes’ classification problem, of which is truthful and cheating groups’ classification. We set 45 test samples for truthful group and cheating group in total. Thus 55 samples are the training samples, of which truthful group contains 16 samples while cheating group contains 39 samples. The performance is revealed in Table III.

The confusion matrix in Table III tells for truthful group, 10 samples are correctly classified while 5 are not; For cheating group, 25 samples are correct and only 5 are wrong. We further get detail accuracy of those 2 categories, 0.667 and 0.833. The average precision of the two groups achieves 0.778, which is enhanced more than 10% from the result 0.644 without feature selection.

D. Discussions

The clues of deceptive behavior proposed in this paper show their effectiveness in the features evaluation. From the baseline accuracy 54%[2], our proposed visual features enhanced the accuracy by 10%. Moreover, among the four lower level features, in single feature evaluation, instantaneous features appear to be more direct than last-for-a-while features. After feature selection and proper classifier design promotion, the final accuracy achieves 77.8% on

TABLE III

THE CONFUSION MATRIX OF CLASSIFYING DECEPTIVE AND TRUTHFUL CASES.

	Truthful	Deceptive	accuracy
Truthful	10	5	0.667
Deceptive	5	25	0.833

average(66.7% truthful recognition rate and 83.3% deceptive recognition rate), which is a leap from the baseline result. Since the truthful and deceptive groups are almost clearly separated, the proposed interactional synchrony feature shows its significance in discriminating deception. Further from Figure.6, the patterns of synchrony feature appear to be different for truthful and deceptive groups. The average feature vector of a group indicates the trend and distribution of all the feature vectors in the same group. Thus, the higher level feature can reflect the degree of synchrony for different groups and further be correctly classified.

Nevertheless, automatic methods can often detect events of synchrony which are missed by the human coders for whatever reason. In particular, we found that the human coders would label a video with no synchrony in it, while our software did detect a number of synchrony events. Hence there is disagreement between the results of the manual analysis and the results of the automatic analysis. Despite a small percentage of false negatives in detecting the events of interest (i.e., nodding, shaking, smiling) the results of the automatic analysis are supportive of the initial hypothesis of the experiment. This means that monitoring synchrony events can be useful for automatic deception detection. False negatives (for shaking and nodding) are attributed to the poor resolution of the input video and the displacement of the facial landmarks was sometimes not large enough to register as a nodding or shaking event.

IV. Conclusions and Future Work

In this paper, we propose an automated framework to analyze interactional synchrony and to investigate how degree of interactional synchrony can indicate whether deceit is present, absent, increasing or declining. This framework includes a robust facial tracking module, an effective expression recognition method, synchrony feature extraction and feature selection methods. Extensive experiments show that our proposed framework is able to robustly analyze interactional synchrony and these synchrony features help to detect deceptions at a reasonable accuracy. In the future, we would like to apply our framework to a greater sample population. We will also further improve our tracking system by incorporating 3D deformable models [14].

References

[1] M. G. Aamodt and H. Custer. Who can best catch a liar?: A meta-analysis of individual differences in detecting deception. volume 15, pages 6–11. The Forensic Examiner, 2006.

[2] C. F. Bond, Jr., and B. M. DePaulo. Accuracy of deception judgements. In *Personality and Social Psychology Review*, volume 10, pages 214–234, 2006.

[3] D. B. Buller and J. K. Burgoon. Interpersonal deception theory. *Communication Theory*, 6:311–328, 1996.

[4] J. K. Burgoon, B. A. L. Poire, and R. Rosenthal. Effects of preinteracton expectancies and target communication on perceiver reciprocity and compensation in dyadic interaction. *Journal of experimental social psychology*, pages 287–321, 1995.

[5] E. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, 52(12):5406 – 5425, Dec 2006.

[6] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[7] N. E. Dunbar, M. L. Jensen, J. K. Burgoon, B. Adame, K. J. Robertson, L. Harvill, and A. Allums. A dyadic approach to the detection of deception. In *HICSS*, 2011.

[8] N. E. Dunbar, M. L. Jensen, K. M. Kelley, K. J. Robertson, D. R. Bernard, B. Adame, and J. K. Burgoon. Cheating and credibility: How modality and power affect veracity detection. In *Annual meeting of the National Communication Association*, 2010.

[9] Y. Freund, R. Iyer, R. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *The Journal of Machine Learning Research*, 4:933–969, 2003.

[10] J. Hirschberg, S. Benus, J. M. Brenier, F. Enos, S. Friedman, S. Gilman, C. Gir, M. Graciarena, A. Kathol, and L. Michaelis. Distinguishing deceptive from non-deceptive speech. In *In Proceedings of Interspeec'2005 - Eurospeech*, pages 1833–1836, 2005.

[11] F. Jiao, S. Li, H.-Y. Shum, and D. Schuurmans. Face alignment using statistical models and wavelet features. In *Proc. CVPR*, pages 321 – 327, Jun 2003.

[12] A. Kanaujia, Y. Huang, and D. N. Metaxas. Tracking facial features using mixture of point distribution models. In *ICVGIP*, pages 492–503, 2006.

[13] T. O. Meservy, M. L. Jensen, J. Kruse, J. K. Burgoon, and J. F. Nunamaker. Automatic extraction of deceptive behavioral cues from video. volume 11 of *ISI'05*, pages 198–208. Springer-Verlag, 2005.

[14] D. N. Metaxas. *Physics-based deformable models: applications to computer vision, graphics, and medical imaging*, volume 389. Springer, 1997.

[15] R. Mihalcea and M. Burzo. Towards multimodal deception detection – step 1: building a collection of deceptive videos. ICMI '12, pages 189–192. ACM, 2012.

[16] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In *Proc. ECCV*, pages 504 – 513, 2008.

[17] L.-P. Morency, I. K. de, and J. Gratch. Context-based recognition during human interactions: Automatic feature selection and encoding dictionary. In *Proceedings of the 10th international conference on Multimodal interfaces*, pages 181–188. ACM, 2008.

[18] K. Prabhakar, S. Oh, P. Wang, G. Abowd, and J. Rehg. Temporal causality for the analysis of visual events. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1967–1974, 2010.

[19] M. Rogers and J. Graham. Robust active shape model search. In *Proc. ECCV*, pages 289 – 312. 2006.

[20] A. A. Salah, J. Ruiz-Del-Solar, C. Mericli, and P.-Y. Oudeyer. *Human Behavior Understanding*. Springer-Verlag, 2012.

[21] J. Shi and C. Tomasi. Good features to track. In *Proc. CVPR*, pages 593 – 600, 1994.

[22] B. E. Turvey. *Criminal profiling: An introduction to behavioral evidence analysis*. Elsevier Academic Press, 2008.

[23] P. Viola and M. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[24] A. Vrij. Why professionals fail to catch liars and how they can improve. volume 9, pages 159–181. Legal and Criminological Psychology, 2004.

[25] C. H. White and J. K. Burgoon. Adaptation and communicative design: Patterns of interaction in truthful and deceptive conversations. *Human Communication Research*, 27(1):9–37, 2001.

[26] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210 – 227, 2009.

[27] F. Yang, J. Huang, and D. Metaxas. Sparse shape registration for occluded facial feature localization. In *Automatic Face and Gesture Recognition*, pages 272 – 277, Mar 2011.

[28] P. Yang, Q. Liu, and D. N. Metaxas. Boosting coded dynamic features for facial action units and facial expression recognition. In *CVPR*, 2007.

[29] P. Yang, Q. Liu, and D. N. Metaxas. Rankboost with l1 regularization for facial expression recognition and intensity estimation. In *ICCV*, pages 1018–1025, 2009.

[30] X. Yu, S. Zhang, Z. Yan, F. Yang, J. Huang, N. Dunbar, M. Jensen, J. Burgoon, and D. Metaxas. Is interactional dissynchrony a clue to deception: Insights from automated analysis of nonverbal visual cues. In *HICSS*, 2012.

[31] S. Zhang, Y. Zhan, M. Dewan, J. Huang, D. Metaxas, and X. Zhou. Sparse shape composition: A new framework for shape prior modeling. In *CVPR*, pages 1025–1032, 2011.

[32] S. Zhang, Y. Zhan, M. Dewan, J. Huang, D. N. Metaxas, and X. S. Zhou. Towards robust and effective shape modeling: Sparse shape composition. *Medical Image Analysis*, 16(1):265 – 277, 2012.