Deception Detection through Automatic, Unobtrusive Analysis of Nonverbal Behavior

Thomas O. Meservy, Matthew L. Jensen, John Kruse, Judee K. Burgoon, and Jay F. Nunamaker Jr., *University of Arizona*

Douglas P. Twitchell, *Illinois State University*

Gabriel Tsechpenakis and Dimitris N. Metaxas, *Rutgers University*

An automated, unobtrusive system identifies behavioral patterns that indicate deception from nonverbal behavioral cues and classifies deception and truth more accurately than many humans.

Every day, hundreds of thousands of people pass through airport security checkpoints, border crossing stations, or other security screening measures. Security professionals must sift through countless interactions and ferret out high-risk individuals who represent a danger to other citizens. During each interaction, the security professional must decide whether the individual is being forthright or deceptive. This task is difficult because of the limits of human vigilance and perception and the small percentage of individuals who actually harbor hostile intent (see the sidebar for more about these challenges). Security personnel can't halt the flow of people and material to extensively gauge the truthfulness of every interaction, so we must do more to help them identify deception and ill intent.

Our research initiative is based on a behavioral approach to deception detection. This approach is appealing for several reasons—foremost because you can use it unobtrusively without the subject's cooperation. For instance, by analyzing a criminal suspect's interview video for movement patterns and comparing it to known deceptive and truthful subjects, you could potentially gain insight into the person's truthfulness. Such an approach might also have direct applications in security screening checkpoints, ticket counters, meetings, speeches, or other situations where deceptive interpersonal communication might occur and security is important.

We attempted to build an automated system that can infer deception or truthfulness from a set of features extracted from head and hands movements in a video. A validated and reliable behaviorally based deception analysis system could potentially have great impacts in augmenting humans' abilities to assess credibility.

Automated deception-detection systems

Automated systems can draw upon a wide variety of potential behavioral indicators of deception. For example, researchers have shown great interest in micro-momentary expressions whereby involuntary, fleeting facial movements reliably suggest deception.[1] Although it doesn't require the polygraph's invasive sensors, this approach is hindered in that it requires unobstructed, high-quality video of the face. To allow flexibility, our approach is designed to pick up more easily recognized movements and behavioral patterns and doesn't rely on minute movements that might be difficult to capture. Figure 1 characterizes three general approaches to deception detection.

By extensively deconstructing deceptive acts, researchers have found that deceivers, in an attempt to retain credibility and deflect suspicion, express patterns of atypical behavior.[2,3] For instance, deceivers often suppress the normal gestures that accompany interaction and appear overcontrolled. Moreover, when they do move, the movement tends to be abrupt. Truth tellers, on the other hand, maintain more smooth and congruent presentations. The discrepancies between the signatures of features extracted from movement on video can be quite telling. Figure 2 illustrates the typical differences between a few simple features extracted from two

1541-1672/05/$20.00 © 2005 IEEE
Published by the IEEE Computer Society

Challenges of Deception Detection

In some highly structured interactions such as financial dealings, we build complex systems with checks and balances to minimize the chances that deception will go unnoticed. In more spontaneous interactions such as security screenings, the costs of constant vigilant monitoring are simply too high. Moreover, most people are subject to myriad biases. Research has revealed that the most common bias is toward simply believing in messages' truthfulness. Security personnel, conversely, might hold the "Othello bias," where they lean toward doubting messages' truthfulness. In any case, when truth and deceit are equally likely, deception detection is about as accurate as random guessing for most humans.[1]

In response, researchers have sought to better understand how deception takes place and how to counter it. There are several accepted approaches, but the most widely known is the physiological approach whereby the deceiver exhibits physical indicators such as increased heart or respiration rates as a result of the arousal associated with deceiving. The polygraph is the most prevalent attempt to detect physiological deception indicators. Although many of the deception-detection approaches to date have demonstrated validity, they often suffer from a level of impracticality. The polygraph, for example, requires controlled situations and extensive cooperation on the part of the subject.

Reference

1. A. Vrij, *Detecting Lies and Deceit: The Psychology of Lying and its Implications for Professional Practice*, John Wiley & Sons, 2000.

short videos (one of a deceiver and one of a truth teller).

Automated systems might be able to help security professionals as they attempt to identify deception in interpersonal exchanges. For systems to infer deception in various real-world situations, they must begin to acquire some of the perceptual and interpretive powers that we take for granted in humans. Researchers in computer vision and artificial intelligence have laid much of the groundwork. Automated systems can recognize size, shape, color, movement, and a host of other attributes that act as a foundation for more nuanced perception.

Our approach

Our approach in creating a system that can identify nonverbal indicators associated with deception or truth is similar to that adopted in many pattern classification problems. In the generic model, the system collects raw

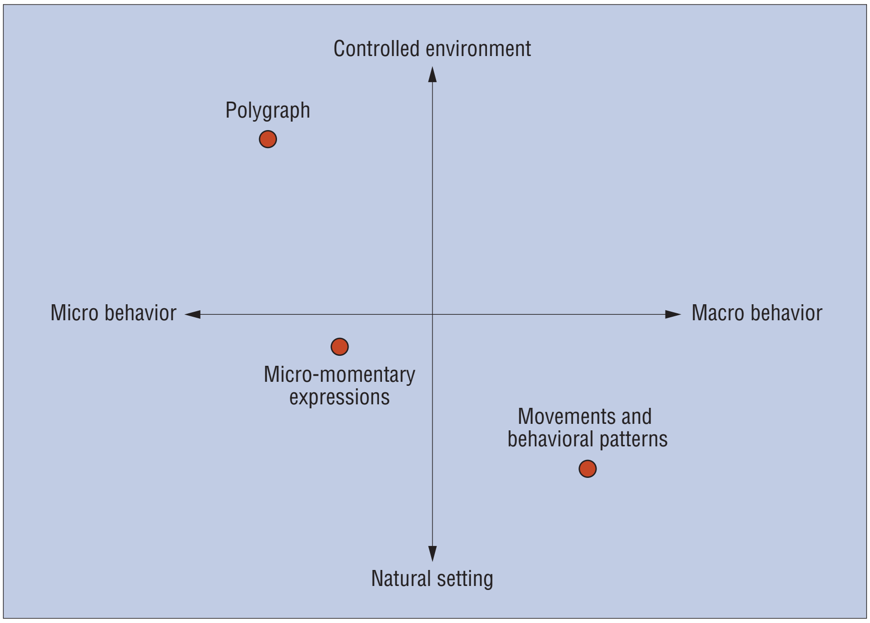


Figure 1. General approaches to deception detection.

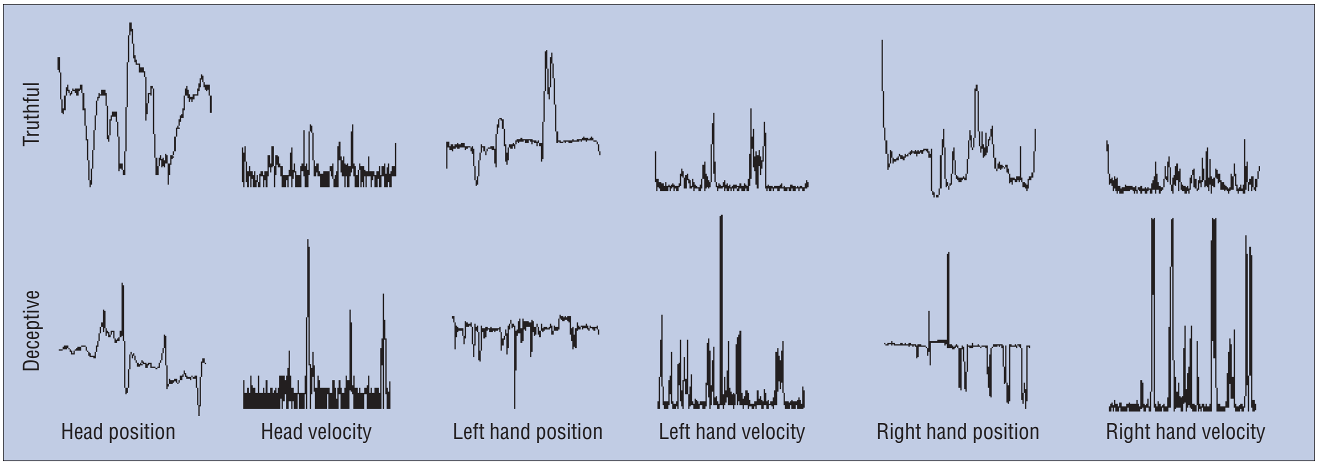


Figure 2. Deceptive and truthful feature signatures.

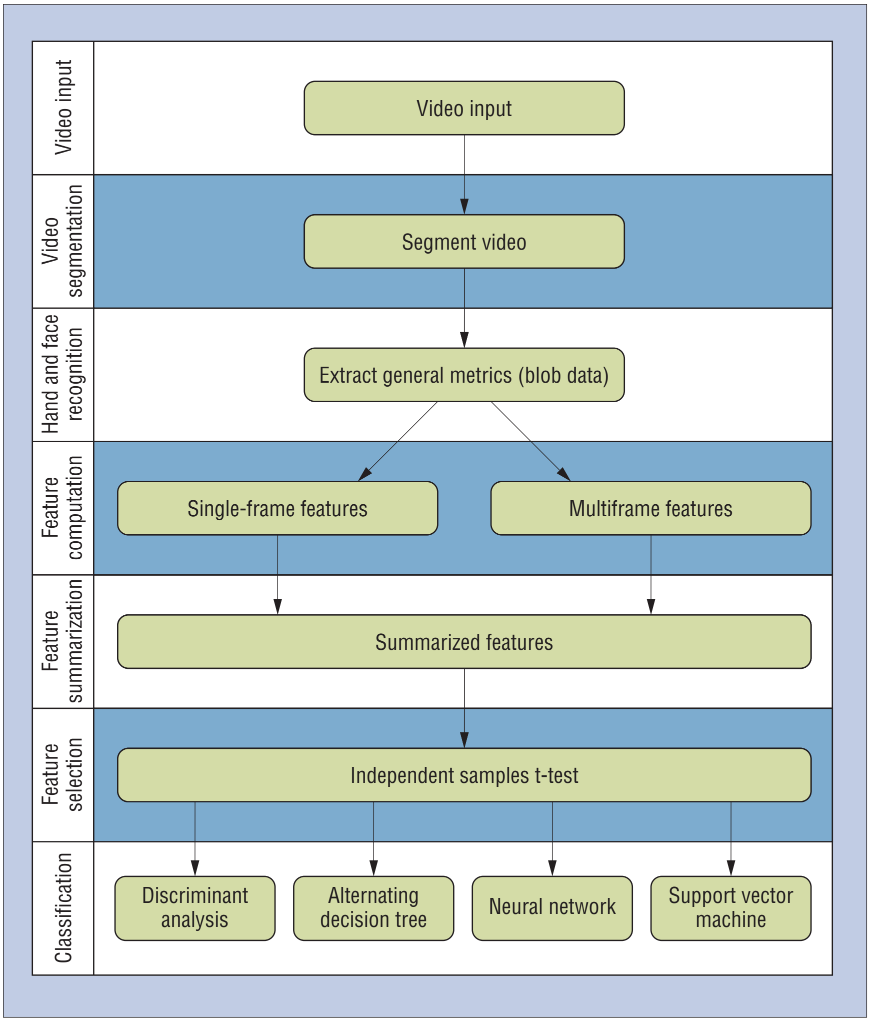


Figure 3. Steps in classifying truth and deception from nonverbal cues.

data and segments it into discrete units. It uses general metrics extracted from these units to compute features. Finally, it uses these features to classify the raw data.

Our system slightly adapts this general pattern. We collect videos of interactions and divide them into meaningful segments. An example of a possible segment might be an answer to a question or a narrative explanation. Our system then analyzes the segmented videos to find the positions of the head and hands in each frame. Using elliptical “blobs” to approximate the location and size of the head and hands, the system calculates the center point, axes’ lengths, and angle of major axis for every blob. It calculates additional features from the basic features extracted from each blob. We can group these features using a taxonomy where the two highest-level categories are single-frame and multiframe features.[4] The system calculates these features for each frame in the video clip. It must then summarize the features so it can classify them using various classification methods. Here we utilize an alternating decision (AD) tree, a neural network, and a support-vector machine (SVM) and compare results with those of discriminant analysis. Figure 3 illustrates the entire process.

Video input

The preferred input for our system is high-quality digital video of a single subject in a sitting position away from any objects, such as a table, that might hide the hands or head. Higher-quality video allows for more reliable position estimation of the head and hands. While high-quality video is optimal, we have also successfully used our system with converted analog video. Significant occlusion and a great deal of change in subject orientation will degrade the results.

Video segmentation

In the prototype system, a system user must manually specify the beginning and ending frames of a segment that the system will review for deception. However, we can envision a future version of the system that automatically segments interesting portions of an interaction.

Hand and face recognition

Although many strategies exist for hand and face recognition, our system uses algorithms developed at Rutgers University’s Computational Biomedicine Imaging and Modeling Center. The recognition algorithms use color analysis, eigenspace-based shape segmentation, and Kalman filters to track the head and hands through each frame of a video segment. Figure 4 shows a sample frame that has undergone blob analysis.

To identify the hands and face in a video sequence, we first use a skin color-identification algorithm that extracts hand and face regions using the color distribution from the image sequence. We prepare a 3D look-up table, which we use for setting the color distribution. We train this 3D LUT, which is based on histogram back-projection, using color sample images; we set it with skin color samples extracted from color images in the video segment. On the basis of the skin color segmentation, we can classify all pixels in the color sequence into either the skin-color region or the background region.

After extracting the hand and face regions from an image sequence, the system computes blobs that could represent the face and hands. This color segmentation process might incorrectly classify a region as a hand or face because it has a color distribution similar to skin color. We use rough searching and fine segmentation to avoid this misclassification. In rough searching, we fit the candidate regions using a simple geometric shape, such as an ellipse. Only the areas that meet such standards remain face and hands candidates.

During fine segmentation, we classify the most face- and hand-like areas as the face and hand. We accomplish this by comparing sample skin samples with subspaces of candidate face and hands in the frame. A feature classifier finds reliable hand and face areas based on an eigenspace that includes subspaces such as face, one-hand, and two-hand. We train this classifier using the samples we used to train the 3D LUT.

To separate hands when they overlap the face, the system computes blobs not only for a single-frame image but also for a time differential image between sequential frames. First, it binarizes the single-frame and time-differential images. We use a Kalman filter to predict each hand's location. The system assigns the blob nearest to the predicted location to either the left or right hand and labels it as a hand blob. The system updates each blob's observed location. If a hand blob exists in the frame, the system updates the observed location using the hand blob. However, if the hand stops in front of the face, no hand blobs originating from the hand appear. Here, the system updates the observed location from the last hand blob's location. Shan Lu and colleagues describe this process in more detail.[5]

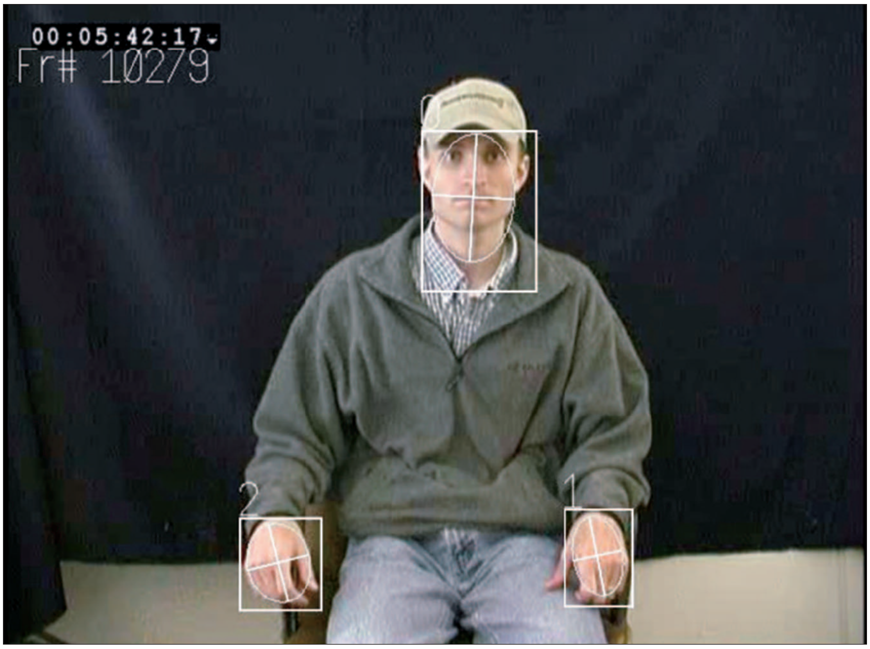


Figure 4. A sample frame in which the hands and face are identified.[4]

Feature computation

From the data each blob provides (see figure 5a), the system calculates a number of additional features. We use a recently proposed, theory-based taxonomy containing nonverbal, movement-based features to discriminate between truthful and deceptive communication.[4]

The system categorizes each feature as a single- or multiframe feature. It calculates single-frame features using information from a single video frame. It can further categorize these features as relational or multirelational features. The relational features represent relationships between two objects—either a blob to another blob or a blob to a reference point or area. In the former case, the system calculates the distance between two blobs of interest using a simple Euclidean distance formula. The distance between the head, the right hand, and the left hand lets us know when all three are touching (distance is zero) or how far apart they are. This feature, which figure 5b depicts, hints at gestures that might indicate nervousness (such as preening, scratching, and rubbing). Figure 5c illustrates the quadrant feature. The system calculates this feature on the basis of an estimation of the width of the shoulders and the bottom of the head blob. This feature lets us understand how often each blob is in each quadrant and can help to discriminate between an open and closed posture.[4] The single-frame, multirelational category contains features that involve information from all three blobs. An example of a single-frame, multirelational feature is the area of the triangle formed by connecting the head and hands blobs.

Multiframe features require information from two or more frames. An example of a multiframe feature is a blob's velocity. We can measure velocity by the distance and direction a blob has traveled between frames. We can calculate the distance using the Euclidean distance formula on the center points of a blob in two successive frames. Figure 6a illustrates the distance feature. We capture direction by calculating the polar coordinate angle of the vector the two center points create. We also convert the polar angle to degrees and then split it into binary-

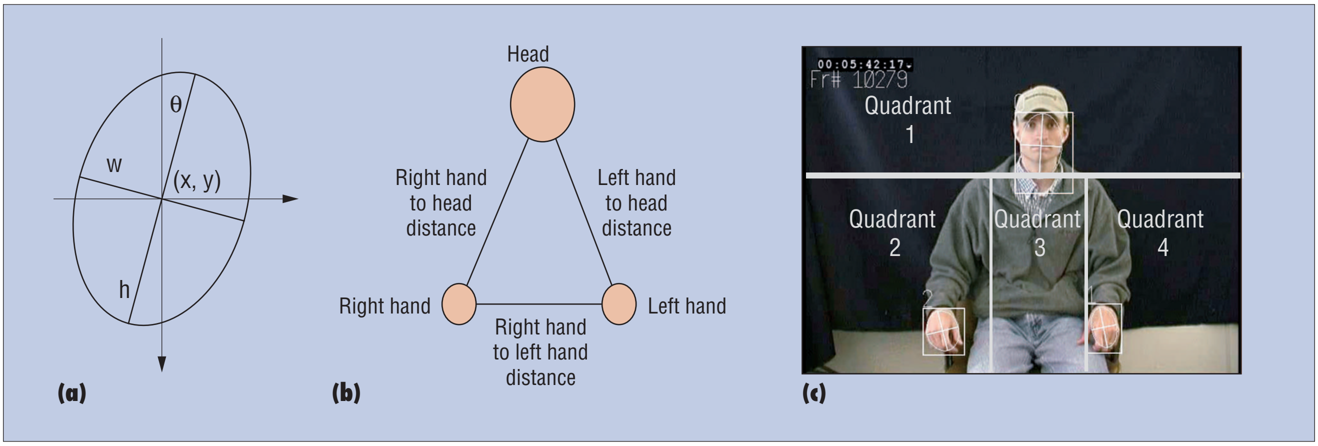


Figure 5. Sample single-frame features:[4] (a) general metrics extracted from each blob; (b) distances between blobs; and (c) quadrant features.

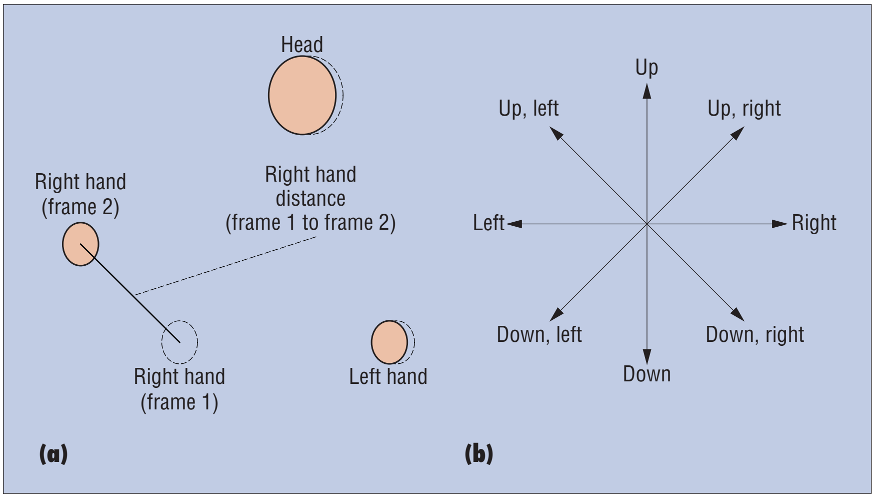


Figure 6. Sample multiframe features:[4] (a) distance between frames and (b) direction of movement.

value classifications. Figure 6b depicts the possible directions of movement.

Feature summarization

The system calculates the features calculated in the previous step for each frame in the video clip. It summarizes single- and multiframe features the blob data provides for every video clip and calculates means and variances of these measurements based on the total number of frames in the video segment.

Feature selection

The complete set of single- and multiframe features contains over 165 features, which focus on the head and hands. Although each feature is meant to single out a particular bit of information, all of these features are based on the general metrics extracted during blob analysis; therefore, they significantly overlap in many cases. We initially included all features when identifying deception. However, models with a reduced feature set performed more accurately, likely because of the relatively small training-set size. Therefore, we adopted a feature-pruning step for the machine classification methods whereby we included features in the machine classification on the basis of statistical significance. This step increases classification accuracy, decreases time requirements for training, reduces noise, and preserves model parsimony.

Classification of truth or deception

Our initial approach in evaluating the features extracted from the video segments was through discriminant analysis. This method is generally accepted as a rigorous technique for differentiating between two groups within a population. For our system's purposes, it provides an effective benchmark with which we can compare the results of the machine classification algorithms.

We chose three machine classification techniques to investigate which method best classifies deception from the selected features. The first classification method we used is an AD tree algorithm.[6] A decision tree is a classification method that structures decision points around the most discriminating features. The AD tree algorithm generalizes both voted stumps and voted decision trees and uses boosting to simplify classification rules.

We also used a neural network multilayer perceptron. The MLP handles continuous data, works effectively even in the presence of errors in the training data, and requires little processing time in classification after training. However, MLP training is time intensive, and node weights are difficult to interpret. Our system uses an MLP with a back propagation learning algorithm.

Finally, we included an SVM classifier. The SVM helps classify complex data because it supports numerous feature vectors and is deterministic. The SVM uses a sequential minimal optimization[7] using scaled polynomial kernels. The SMO algorithm breaks the training problem into a series of small quadratic programming problems. The training problem's decomposition allows for a potentially large training set. Once the small quadratic programming problems are solved, the SVM can classify a population.

Investigating classification methods for deception detection

To test the classification techniques' detection ability, we constructed an experiment using videos of deceptive interactions. The data set's limited size and scope precludes drawing many broad, generalizable lessons; however, it does afford the opportunity to establish the concept's viability. The videos in the data set came from a mock theft experiment designed to reveal cues that can be used in deception detection. The experiment's design directed randomly selected undergraduate communication students in a large Midwestern university to steal a wallet that was left unattended in a classroom. Other students who participated were present during the theft but didn't take the wallet. Trained interviewers later interviewed all participants via chat, audio, and face-to-face channels and recorded these interactions.

We used the face-to-face interactions from this experiment to test our prototype system. We could have included a total of 42 possible face-to-face interactions in the study; we didn't use four of them because of technical problems with the video work or because the participant didn't follow instructions. Each interaction consisted of several question-answer exchanges. In this study, we included only the narrative regarding the actual theft in the analysis. Of the 38 responses, 16 were truthful and 22 were deceptive.

Feature selection

We reduced the number of features used in the discriminant analysis and machine classification techniques by independent samples t-tests across deceptive and truthful states. We used all features whose means were significantly different ($p < .05$) between the truthful and deceptive conditions. The reduced feature set contained five features including the mean difference in head angles between frames, the mean difference in right hand angles between frames, the mean difference in left hand angles between frames, variance of the head y position, and variance of the number of frames the left hand is in quadrant 4. You could use other methods in feature reduction; however, simple t-tests were the most straightforward, and others have used them successfully in automated deception-detection systems.[8] The feature-selection strategy has a tremendous impact on deception-detection ability. Therefore, we plan to explore additional, more effective ways to reduce the number of features used for each classification method.

Classification methods

We performed a discriminant analysis with the five selected features, and the model was not significant ($p = .058$). We evaluated the AD tree, SVM, and MLP in a Java environment using the Weka library.[9] The AD tree had a maximum depth of 3 with 28 total nodes and 19 predictor nodes. The SVM used an ADA boosting algorithm to enhance performance. The MLP handled the five single- and multiframe features as inputs and had two output nodes (deception, truth) with four hidden layer nodes.

Table 1. Cross-validated classification results.

Classification method	Overall correct classification (%)	Class	Precision	Recall
Discriminant analysis	55.3	Guilty Innocent	0.609 0.467	0.636 0.438
Alternating decision tree	57.8947	Guilty Innocent	0.65 0.5	0.591 0.563
Neural network	71.0526	Guilty Innocent	0.824 0.619	0.636 0.813
Support vector machine	71.0526	Guilty Innocent	0.667 1	1 0.313

Classification results

Table 1 contains cross-validated classification results. The discriminant analysis we discuss uses a hold-one-out cross-validation strategy. To approximate this strategy for the other methods, we used a 38-fold cross-validation model (as $N = 38$).

The overall correct classification is the percentage of testing instances that the system classified correctly. A naïve or baseline classifier that assumed that all instances in the training set should be considered guilty would attain 57 percent accuracy. The SVM and the neural network produced the highest accuracy at 71.1 percent. We further supported the SVM's performance by running 20 repeats of 10-fold cross-validation and performing a t-test comparing performance accuracy. The SVM was significantly better than the baseline classifier at $\alpha = 0.10$. However, none of the other classifiers were significantly better than the baseline. When cross-validated, neither the AD tree nor the neural network did better than the baseline. Therefore, an SVM might be an effective classifying technique in identifying deception from video cues. When an SVM is used, detection accuracy appears to rise above that which most humans exhibit. A larger data set (and therefore training set) is needed to provide stronger support for this conclusion.

Precision and recall are measures specific to each class. For example, in the discriminant analysis, you can think of the precision for the guilty class as the probability that the instances that the system classified as guilty are actually guilty. You can think of recall for the guilty class simply as the number of identified guilty clips divided by the total number of guilty clips. When cross validated, the SVM had a precision of 0.667 and perfect recall in the guilty condition. However, the SVM's performance indicates that it would have a high false-positive rate. The neural network produces a lower false-positive rate but also classifies more guilty individuals as innocent. In some security situations, such as transportation screening or border security, a high false-positive rate would result in a greater number of unnecessary secondary screenings and investigations. At the same time, it would reduce the likelihood of those with ill intent passing through.

In spite of this system's potential uses, many technical and utilization challenges must be overcome before it can be fielded for homeland security.

System use

This technological approach to detecting deception in video has widespread potential applications. Actual system use will depend on a number of variables that aren't entirely clear. The most important factors are the system's validity and reliability. In addition, questions exist about cost, system speed, training, and privacy.

The use with the greatest potential applicability is the personal interview. For example, visa interviews take place at embassies and consulates in large numbers throughout the world. The process is often slow and arduous, and neither the interviewers nor the interviewees are pleased with it. By providing the interviewers with reliable tools to help recognize deceivers, the interviews can become more efficient and effective.

Another high-payoff system application would be at security checkpoints. At airports, sporting events, government buildings, and borders, people are subjected to varying levels of searches and questioning about their purposes and possessions. An automated system that could identify suspicious body movements and alert security personnel would be an improvement.

Beyond the attended system, there's also great potential for fully automated systems that could begin to approximate human behavior for specific tasks. For instance, intelligence analysts have access to video libraries that go largely ignored because analyzing video is so labor intensive. An automated system could easily pore through thousands of hours of video looking for and alerting analysts to specific indicative behaviors.

System limitations

In spite of this system's potential uses, many technical and utilization challenges must be overcome before it can be fielded for homeland security. Quite possibly the greatest limitation is that a behavioral system's ability to detect deception is rooted in the need to elicit the same types of deception across individuals. This often requires extensive interaction that might not be possible at a security checkpoint or similarly short interaction. Although it might be less of a limitation in longer, more deliberate interviews, the interviewer still needs to probe enough to force outright deception.

Another issue we've identified is the level of deception detection's granularity. Ultimately, we'd like to be able to identify deception on a statement-by-statement basis. Deception is complex and strategic, and deceivers often intersperse truth with lies or build lies on a foundation of truths. This system is trained

The Authors

Thomas O. Meservy is a research associate at the Center for the Management of Information at the University of Arizona and is pursuing a PhD in management information systems with a minor in cognitive science from the University of Arizona. His research interests include pattern classification, knowledge representation and sharing, and intelligent agents and multiagent systems. He holds a masters in information systems management from Brigham Young University. He is a member of the IEEE. Contact him at the Univ. of Arizona, Center for the Management of Information, McClelland Hall Rm. #427, 1130 E. Helen St.,Tucson, AZ 85719-4427; tmeservy@cmi.arizona.edu.

Matthew L. Jensen is a research associate at the Center for the Management of Information at the University of Arizona and is pursuing a PhD in management information systems with a minor in management and policy from the University of Arizona. His research interests include applied machine learning, computer-aided decision making, and deception detection. He holds a masters in information systems management from Brigham Young University. He is a member of the IEEE. Contact him at the Univ. of Arizona, Center for the Management of Information, McClelland Hall Rm. #427, 1130 E. Helen St., Tucson, AZ 85719-4427; mjensen@cmi.arizona.edu.

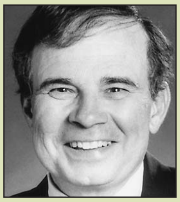
John Kruse is the director of systems development at the Center for the Management of Information at the University of Arizona. He is working to develop learning and language processing software for deception and intent detection. He received his PhD in management information systems with a minor in management and policy from the University of Arizona. Contact him at the Univ. of Arizona, Center for the Management of Information, McClelland Hall Rm #427, 1130 E. Helen St.,Tucson, AZ 85719-4427; jkruse@cmi.arizona.edu.

Douglas P. Twitchell is an assistant professor at Illinois State University. His research interests include text mining, conversational analysis and profiling, machine learning, and natural language processing. He received his PhD in management information systems from the University of Arizona. He is a member of the IEEE. Contact him at the School of Information Technology, Illinois State Univ., Campus Box 5150, Normal, IL 61790-5150; dtwitch@ilstu.edu.

Gabriel Tsechpenakis is a postdoctoral researcher at the Center for Computational Biomedicine Imaging and Modeling of the Division of Computer and Information Sciences at Rutgers University. His research interests are in computer vision and machine learning, specifically 2D and 3D tracking of deformable and articulated objects, American Sign Language recognition, and nonverbal behavior analysis for deception recognition from visual cues. He has a PhD in computer engineering from the National Technical University of Athens. He is a member of the IEEE, TCG (Greek Technical Chamber), and HAMEE (Hellenic Association of Mechanical and Electrical Engineers). Contact him at the Division of Computer and Information Sciences, Rutgers Univ., 110 Frelinghuysen Rd., Piscataway, NJ 08854-8019; gabrielt@research.rutgers.edu.

Judee K. Burgoon is a professor of communication, a professor of family studies and human development, and the director of human communication research for the Center for the Management of Information at the University of Arizona. Her research interests are in deception, trust, interpersonal interaction, and new technologies. She received her PhD in communication and educational psychology from West Virginia University. She is a member of the International Communication Association, the National Communication Association, the Society for Experimental Social Psychology, and the Society for Personality and Social Psychology. Contact her at the Univ. of Arizona, Center for the Management of Information, McClelland Hall Rm. #427, 1130 E. Helen St., Tucson, AZ 85719-4427; jburgoon@cmi.arizona.edu.

Dimitris N. Metaxas is a professor in the Division of Computer and Information Sciences and a professor in the Department of Biomedical Engineering at Rutgers University. He also directs the Center for Computational Biomedicine, Imaging, and Modeling. His research interests are in the simultaneous segmentation and fitting of complex objects, shape representation, deterministic and statistical object tracking, and gesture recognition. He has a PhD in computer science from the University of Toronto. He's a member of the IEEE and ACM. Contact him at the Division of Computer and Information Sciences, Rutgers Univ., 110 Frelinghuysen Rd., Piscataway, NJ 08854-8019; dnm@cs.rutgers.edu.



Jay F. Nunamaker Jr. is the Regents and Soldwedel Professor of MIS, computer science, and communication and the director of the Center for the Management of Information at the University of Arizona, Tucson. He received his PhD in systems engineering and operations research from Case Institute of Technology. He is a member of the IEEE and ACM. Contact him at the Univ. of Arizona, Center for the Management of Information, McClelland Hall Rm. #427, 1130 E. Helen St., Tucson, AZ 85719-4427; jnunamaker@cmi.arizona.edu.

on short responses to a single question. Nonetheless, even within that response, varying levels of truthfulness exist. We can only assess the entire segment.

Technically, many issues bound this system's applicability; foremost are the requirements for video quality. The color segmentation system compels the use of high-quality video equipment and good lighting. However, the video quality requirements aren't as high as those for methods that analyze micromomentary facial expressions. Additionally, the subject's orientation and framing are important. We can derive little information if the hands and head aren't in view. Finally, the system as currently configured doesn't approach the speeds and ease of use necessary for widespread employment.

Even with significant technical challenges in mind, we believe that this behaviorally based video deception-detection system demonstrates preliminary potential for homeland security applications. Although we can't expect any system to make perfectly accurate judgments on something as complex as deceptive behavior, we anticipate relevant improvements. By augmenting security professionals, we can expect to make them more effective and efficient as systems alert humans to possible risks. Automated systems might even become reliable enough to replace humans in certain circumstances, thus allowing a redistribution of human assets. ■

Acknowledgments

Portions of this research were supported by funding from the US Air Force Office of Scientific Research under the US Department of Defense University Research Initiative (Grant #F49620-01-1-0394) and by the US Department of Homeland Security (Cooperative Agreement N66001-01-X-6042). This article's views, opinions, and findings are those of the authors and shouldn't be construed as official Department of Defense or Department of Homeland Security positions, policies, or decisions.

References

1. M.G. Frank and P. Ekman, "The Ability to Detect Deceit Generalizes across Different Types of High-Stake Lies," *J. Personality and Social Psychology*, vol. 72, no. 6, 1997, pp. 1429–1439.
2. M. Zuckerman, B. DePaulo, and R. Rosenthal, "Verbal and Nonverbal Communication of Deception," *Advances in Experimental Social Psychology*, vol. 14, L. Berkowitz, ed., Academic Press, 1981, pp. 1–59.
3. B. DePaulo et al., "Cues to Deception," *Psychological Bull.*, vol. 129, no. 1, 2003, pp. 74–118.
4. T.O. Meservy et al., "Automatic Extraction of Deceptive Behavioral Cues from Video," *Intelligence and Security Informatics*, LNCS 3495, Paul Kantor et al., eds., Springer-Verlag, 2005, pp. 198–208.
5. S. Lu et al., "Blob Analysis of the Head and Hands: A Method for Deception Detection," *Proc. 38th Ann. Hawaii Int'l Conf. System Science* (HICSS 05), IEEE CS Press, vol. 1, no. 1, 2005, p. 20c.
6. Y. Freund and L. Mason, "The Alternating Decision Tree Learning Algorithm," *Proc. 16th Int'l Conf. Machine Learning*, Morgan Kaufmann, 1999, pp. 124–133.
7. J. Platt, *Fast Training of Support Vector Machines using Sequential Minimal Optimization*, MIT Press, 1998.
8. T. Qin, J.K. Burgoon, and J. Nunamaker, "An Exploratory Study on Promising Cues in Deception Detection and Application of Decision Tree," *Proc. 37th Ann. Hawaii Int'l Conf. System Sciences* (HICSS 04), IEEE CS Press, 2004, http://csdl2.computer.org/comp/proceedings/hicss/2004/2056/01/205610023b.pdf.
9. I.H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools with Java Implementations*, Morgan Kaufmann, 2000.