Identifying Single-Ended Contact Formations from Force Sensor Patterns

Marjorie Skubic and Richard A. Volz

Abstract— **We present two methods of rapidly (less than 1 ms.) identifying contact formations from force sensor patterns, including friction and measurement uncertainty. Both principally use force signals instead of positions and detailed geometric models. First, fuzzy sets are used to model patterns and sensor uncertainty; membership functions are generated automatically from training data. Second, a neural network is used to generate confidence levels for each contact formation. Experimental results are presented for both classifiers, showing excellent results. New insights into the data sets are discussed, and a modified training method is presented which further improves the performance. The classification techniques are discussed in the context of robot programming by demonstration.**

Keywords— **contact formation, force sensing, classifier**

I. INTRODUCTION

A key attribute of robots is their programmability. However, programming difficulty remains a critical barrier, especially in tasks such as assembly operations, involving contact between the robot and its environment. Uncertainties further exacerbate the problem. Generally, programming of contact tasks relies on precise positioning of the workpiece, often achieved by specialized fixtures. New tasks may require new parts and fixtures, making acquisition of such precise information for each task difficult.

We believe that combining robot programming by demonstration with modeling the assembly task as a sequence of discrete states (contact formations [1]) can simplify the programming process and make the program more robust to discrepancies in position and orientation. The utility of this approach was demonstrated by Kosuge et al [2]. However, the methods used to identify the discrete states require detailed geometric models (e.g., [3], [4]), and even ignoring uncertainty and friction, the process can be complex. Further, ambiguities may inhibit correct identification.

We present an efficient, sensor-based scheme for identifying contact formations without detailed geometric models that operates in the presence of measurement uncertainty and friction. Our method uses force sensor data only. While we believe that using multi-modal information, e.g., vision, force, voice, and geometric information may ultimately lead to superior performance, we have concentrated on force signals to see what can be done with them alone. We hope to gain insight on how to combine force information with geometries and other sensing modalities. The successful use of this force-based approach to learn and complete simple manipulation tasks is described in [5], [6].

This work has been supported by NSF under grants CDA-9115123, CDA-9422123, and EID-9017249 and by the State of Texas under TATP grant 999903-095.

M. Skubic is with the Computer Engineering and Computer Science Dept., Univ. of Missouri-Columbia, Columbia, MO 65211. R. Volz is with the Dept. of Computer Science, Texas A&M Univ., College Station, TX 77843-3112

II. BACKGROUND

The contact formation (CF) was proposed by Desai and Volz [1] as a qualitative discrete state to describe how 2 or more objects are in contact. The original use targeted generation of assembly programs from CAD models. As such, most CF identification methods used data from detailed geometric models, and position and force sensors. [1] identified a CF by formulating a hypothesis and verifying it with static equilibrium equations, using active force sensing to resolve ambiguities.

Hirai and Asada derived classifiers from the geometric model using polyhedral convex cones (PCCs) [3]. The ranges of possible forces/moments and displacements measureable at each CF were represented as a union of PCCs. Discriminant functions provided by the PCC face vectors were used to determine the CF. Sensor uncertainties and friction were not considered. Recently, Mosemann et al extended this work to include a static friction model and noted that the addition of friction made a significant difference in correct CF classification [7].

Farahat, et al considered friction and sensor noise where measured forces overlap more than one CF [4]. Using geometric models, they solved a linear program to determine CF feasibility, and ranked feasibility by the distance between the measured force and the cone of allowable forces. However, solving the linear program was found to be too slow for real time.

McCarragher and Asada used rigid body dynamics to identify CFs [8]. Constraints were added with Lagrange multipliers and a velocity constraint matrix. Qualitative states (QS – defined as a unique combination of positive, negative, or zero values for position, velocity, and acceleration) were enumerated. Templates, precalculated as a sequence of QSs for contact motion, were used to identify the CF sequence during assembly. But thresholds identifying the QSs can be difficult to find, and geometric models are needed

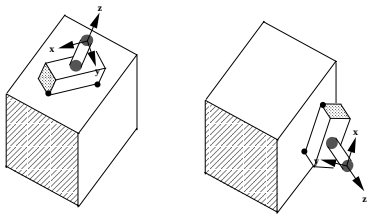

Fig. 1. An Example of Two Contact Formations which Belong to the Same Class of Single-Ended Contact Formations

to precalculate the templates. Recent refinements use linear discriminant functions instead of thresholds [9].

Hovland and McCarragher proposed FFTs for capturing the dynamics of contact changes and Hidden Markov Models to model the resultant information [10]. While they achieved a 97% success rate, they reported times of 0.5-0.6 seconds. Also, significant training is required for each contact transition.

Hara and Yokogawa [11] used fuzzy sets to recognize CFs; however, only a small number of CFs were considered and no general method was shown. Cervera et al have proposed a self-organizing neural network which maps force signals to a 2D grid [12]. This work introduces the idea of clustering force signals for state identification. However, our work includes a critical preprocessing step which minimizes sensor ambiguities.

III. SINGLE-ENDED CONTACT FORMATIONS

A. Characterization

Contact formations [1] provide a qualitative description of how 2 or more objects contact each other (e.g., edge 1 of one object touches side B of another). In contrast, single-ended contact formations (SECF) provide a one-sided description of how a grasped object touches its environment (e.g., edge 1 of a grasped object touches any side in another). Figure 1 gives an example of 2 contact formations but the same SECF.

Our goal is to recognize the SECFs from force data only, without using geometric or position information, and in the presence of measurement uncertainty and friction. While it is not possible to determine the complete contact formation from force data alone, it may be possible to determine a SECF from such data. This is the haptic information a human would use. We approach the problem experimentally. Force data is collected for a set of known SECFs and used as a basis for developing classifiers to identify SECFs in real time.

B. Sensory Patterns

Essential to building a classifier is existence of sensory features that distinguish SECFs. To investigate this, force sensory data were collected for a set of known SECFs. Several test objects were used; the data shown in Figure 3 were collected using the small,

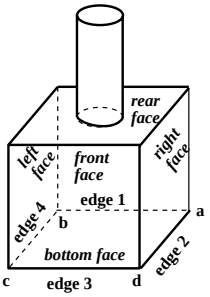

Fig. 2. Test Workpiece

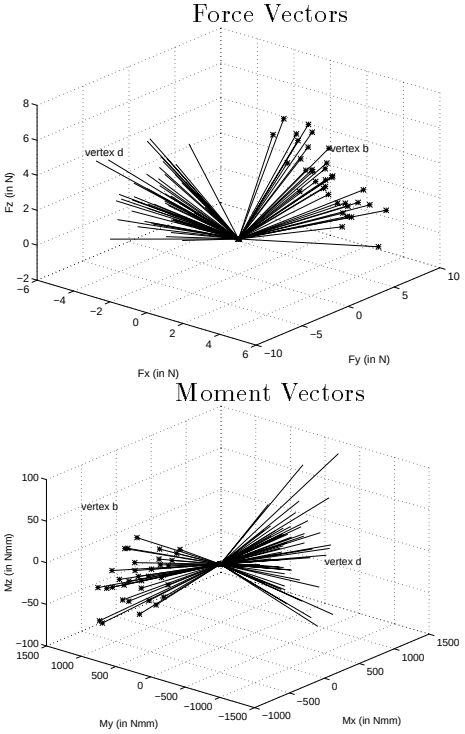

Fig. 3. Force and Moment Vectors of 2 Vertex SECFs

plastic test block in Figure 2. The block was held in a stationary configuration by the robot arm while a flat plate was moved manually. Signals were collected from a wrist force sensor. The volume of data required is primarily related to the number and similarity of SECFs which might occur during operation. For each SECF, representing the range of allowable force vectors is important, not the sheer number of samples. Table I shows the number of samples taken per object.

Figure 3 shows the force and moment vectors (in the sensor frame) for two data sets, which reveal cone-shaped patterns. Factors affecting the pattern include the geometric shape, motion constraints, friction, uncertainty due to sensor noise, and errors in data collection. As a reduced sampling was used for clearer presentation, the figure does not show the full range of allowable forces, but does reveal distingushing patterns. These cone-shaped patterns are consistent with

theoretical discussions in [3] and [4], which are based on geometric relationships. However, in the theoretical approach, it is difficult to account for real-world effects such as sensor noise, behavior due to friction, and workpiece deformation and imperfection. By using a sensor-based approach, we hope to account for these difficult-to-model effects.

IV. CLASSIFICATION OF SINGLE-ENDED CONTACT FORMATIONS

We develop two classifiers that convert a force sensor reading into a SECF identification and confidence level indication. The first handles uncertainty using fuzzy set theory, while the second uses a neural network to generate SECF confidence levels.

Each classifier has one logical rule per class, which describes the mapping between sensory data and SECF. Let F_i and M_i be the sets of possible force and moment vectors for class C_i. Then the rule for each C_i class has the following form:

if *force* is in the F_i set and
moment is in the M_i set
then *class* is C_i

where *force* and *moment* are the sensed vectors.

The force and moment vectors are normalized, yielding projections onto the force and moment unit spheres. The resulting projected clusters reflect the shape and size of the cone patterns.

A. The Fuzzy Classifier

In this classifier, fuzzy sets are used to deal with the consequences of an imperfect force sensor (noise) and other real world uncertainties. The logical rules are expanded into fuzzy rules using the 6 normalized sensor signals as inputs. For each normalized sensor component, a set of fuzzy membership functions is generated from a set of training signals. Let f_x, f_y, f_z, m_x, m_y, and m_z be the normalized sensor components. The membership functions for each class C_i are denoted by $\tilde{F}x_i$, $\tilde{F}y_i$, $\tilde{F}z_i$, $\tilde{M}x_i$, $\tilde{M}y_i$, and $\tilde{M}z_i$. The expanded rule for C_i is as follows:

if f_x is $\tilde{F}x_i$ and f_y is $\tilde{F}y_i$ and f_z is $\tilde{F}z_i$ and
m_x is $\tilde{M}x_i$ and m_y is $\tilde{M}y_i$ and m_z is $\tilde{M}z_i$
then *class* is C_i

For each proposition in the antecedent, the grade of membership is calculated (e.g., the degree to which f_x belongs in the $\tilde{F}x_i$ set), yielding a number in [0,1]. The membership grades for each proposition are combined using fuzzy conjunction; the result is interpreted as the confidence level of being in the specified class. The confidence levels are calculated for each SECF; the highest level is the identified SECF class. This resolves classification ambiguities.

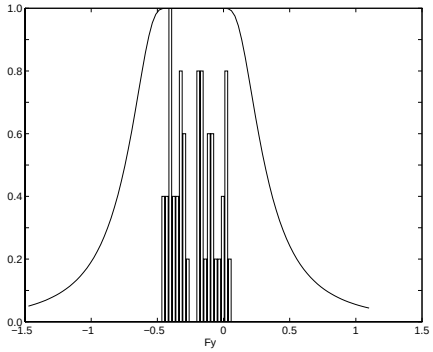

Fig. 4. The Distribution of the Edge 2 (Fy) Training Data Shown with the Generated Membership Function

The membership functions are generated automatically using supervised learning. Training data is acquired by demonstrating each SECF class and collecting the force and moment signals generated. For each sensor component, j, in each class C_i the mean, μ_{ij}, and standard deviation, σ_{ij}, are calculated. These are used as parameters to generate the membership functions, $\pi_{ij}(r_j)$, as shown below:

$$\pi_{ij}(r_j) = \begin{cases} 1 - e^{-(3\sigma_{ij}/|\mu_{ij}-r_j|)^3} & \text{if } r_j \neq \mu_{ij} \\ 1 & \text{if } r_j = \mu_{ij} \end{cases}$$

This function shape was chosen to capture the possible distribution of component values in a class, as well as sensor uncertainties. Figure 4 shows the distribution for a representative sensor component with the generated membership function. Various factors contribute to the range in possible force signals for an SECF. Unconstrained degrees of freedom that allow a workpiece to move and still maintain the desired SECF may result in force component changes. Additionally, friction and sensor uncertainty can cause force changes. The flat portion of the membership function allows for this variation. Near the boundary areas, the curve drops off, indicating a reduced SECF confidence level.

The Hamacher product [13] is used as the fuzzy conjunction operator; for 2 sets, it is defined as shown:

$$H(\pi_a(r_a), \pi_b(r_b)) = \frac{\pi_a(r_a) * \pi_b(r_b)}{\pi_a(r_a) + \pi_b(r_b) - \pi_a(r_a) * \pi_b(r_b)}$$

It can be used in an iterative fashion, until the membership functions of all the components are combined, yielding a Hamacher product $H_i(\pi_{i1}(r_1), ..., \pi_{i6}(r_6))$. [1]

The minimum function and algebraic product were also studied as conjunction operators. Figure 5 shows a comparison; the Hamacher and algebraic products provide better discrimination among classes that are

[1] The order in which the features are combined is irrelevant, as the Hamacher product follows the fuzzy t-norm requirements for commutativity and associativity.

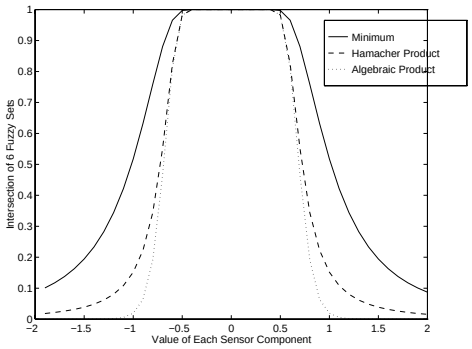

Fig. 5. A Comparison of 3 Conjunction Operators, Showing the Result of Combining 6 Identical Membership Functions, where $\mu = 0.0$ and $\sigma = 0.3$

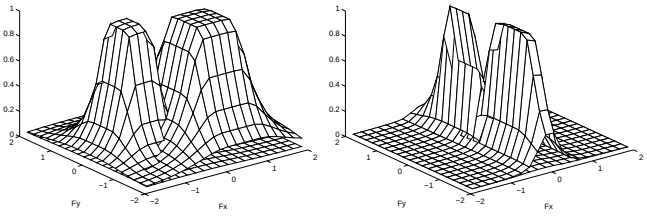

Fig. 6. Combinations of Two Membership Functions for Two Vertices (left) and Two Edges (right)

close. The Hamacher product was chosen as it consistently yielded the highest success rate for this set of membership functions and sensor signals.

The effects of the combination can be seen in Figure 6, where 2 of the 6 components are combined. The distinct shapes are apparent. Figure 6 illustrates the case where the edges are aligned with a sensor axis. Results in Section V will show that when they are misaligned, regions are not as clearly separated. We also note that this classifier combines 6 serially computed, one-dimensional membership functions, instead of using multivariate functions and hence may have limitations. The use of the neural network, presented in the next section, is intended to overcome this limitation.

B. The Neural Network Classifier

The neural network (NN) classifier learns a generalized mapping of force signals to SECF confidence levels. It is derived from a NN mapping force signals to velocity commands proposed by Asada [14]. The generalized architecture is shown in Figure 7, where the inputs are the normalized forces and moments. The number of output nodes is the number of SECF classes. The output values are interpreted as confidence levels; the highest level indicates the identified class.

The number of hidden nodes is based on the coarse geometric shape of the object, one for each vertex in some SECF. Surface and edge contacts are modeled as a finite set of point contacts, e.g., an edge is modeled as 2 point contacts. The connections between the inputs and the hidden nodes correspond physically to the forces exerted on the object's vertex points. The final layer then maps this to the corresponding SECF.

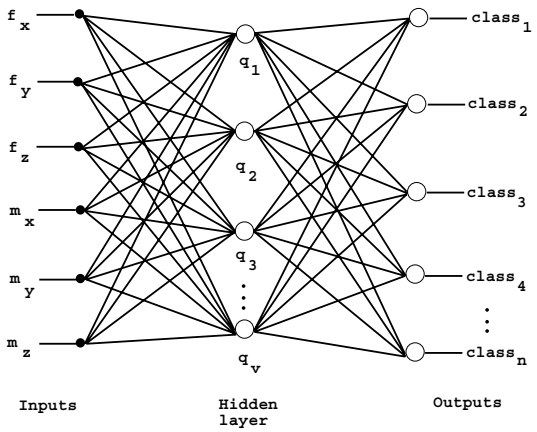

Fig. 7. The Generalized Architecture for n Classes and v Vertices

Training data is again acquired by demonstrating each SECF and collecting force signals. NN training is accomplished using backpropagation with a constant learning rate (LR). The bipolar sigmoid is used as the activation function for both hidden and output nodes, as is typical for learning nonlinear mappings [15]. Training is continued until the total squared error (TSE) establishes an increasing pattern. The final weights and biases are those with the smallest TSE.

C. No Contact Case

While not critical to the methodology presented, in reality, one must deal with situations in which there is no contact. We address this in a simple manner using a force threshold. In the case of the neural net classifier, the force magnitude, $|f|$, is compared with a threshold level ϵ. In the case of the fuzzy classifier, we use a fuzzy threshold membership function, $NOT\tilde{Z}ero$, as shown:.

$$NOT\tilde{Z}ero(|f|) = \begin{cases} e^{-(3\sigma/|f|)^3} & \text{if } f \neq 0 \\ 0 & \text{if } f = 0 \end{cases}$$

V. EXPERIMENTAL RESULTS

Both classifiers were trained using supervised learning with data sets generated by demonstrating a known SECF. A JR3 wrist force sensor provided the data. It was found that data from an unintended SECF could occur, thus generating a kind of noise. Although every effort was made to collect clean data, undoubtedly some noise was included in the data. For each set collected, data were separated into 2 equal groups, one used for training and the second used for testing. Results are shown for the testing data.

A. Data Sets

Six sets of training data were used, representing different object shapes and SECF combinations.

- The *pentagon-11* set contains 11 classes and was made using a plastic, pentagon-shaped block. The set contains 6 surface and 5 edge SECFs.
- The *plug-7* set used a slightly flexible 3-prong electrical plug. The 7 classes included single, double, and triple prongs contacting a flat surface.
- The *peg-11* set used a 4-sided aluminum peg. The 11 classes included the bottom, 4 side surfaces and 4 bottom edges. 2 were created by jamming the peg against opposing surfaces of a hole. The peg was rotated 40 degrees so its edges did not align with the sensor frame.
- The *peg-9* set used the aluminum peg. The 9 classes included the bottom, 4 side faces, and 4 double combinations (e.g., front & left), to test multiple constraints.
- The *square-17* set used a small, plastic, square block. The 17 classes included the bottom and 4 side surfaces, 8 double-sided and 4 bottom corner surface combinations (e.g., *right front bottom* surfaces).
- The *square-8* set also used the square block. It was designed to test the situation in which 2 or more SECFs could produce the same force, but not necessarily the same moment, vectors (e.g., *right front edge* vs. *right front surfaces*). The 8 classes include the 4 bottom edges and 4 combinations of side and bottom surfaces.

B. Classifier Results

First, to determine whether any classification can be made at all, both classifiers use a confidence threshold. If no SECF confidence level passes, then the data vector is said to be unclassified. In this work, the confidence threshold was empirically set to 0.5 for both. For noisier sensors or environments with more uncertainties, the confidence level could be relaxed.

B.1 Fuzzy Classifier

The fuzzy classifier results are shown in Table I. The performance for the *pentagon-11* and *square-17* sets is quite good. For the *peg-11* and *peg-9* sets, the fact that the edges of the peg were not aligned with the sensor frame may contribute to the decreased performance.

TABLE I
FUZZY CLASSIFIER RESULTS

set	good	misses	unclass.	%success
pentagon-11	1822	15	8	98.8
plug-7	168	5	6	93.9
peg-11	1333	82	6	93.8
peg-9	3354	204	26	93.6
square-17	8233	52	166	97.4
square-8	1657	299	28	83.5

An interactive visualization program was developed to display the overlap in the force and moment sets. As only three dimensions can be easily interpreted at a time, the force vectors were displayed in one rotatable window and the moments in another. If there is no overlap in one or the other of the sets, we can be sure of separation in 6 dimensions. If there is overlap in both, one cannot tell, and further analysis is required.

For the *plug-7* set, a few vectors looked misplaced, probably as a result of the training-generated noise. Given the smaller number of training vectors for *plug-7*, this could skew the means and standard deviations

No overlap was seen in the *pentagon-11* sets. In the remaining four data sets, there was apparent overlap. The *peg-11*, *square-17*, and the *square-8* sets contained significant overlap or very close sets between some pairs of SECF classes; *square-8* showed particularly strong overlap in both force and moment, and the classifier results reflected this. The *peg-9* set contained only modest overlap. The overlap in the peg sets, may have contributed to the decreased performance.

B.2 Neural Network Classifier

Table II shows the NN classifier results. While several results are quite good, the *peg-11*, *peg-9*, and *square-17* performance, where overlap observed, was worse. Remarkably, the NN classifier performed much better for *square-8*, suggesting that while there was overlap individually in the force and moments sets, in 6 D there was separation that the NN recognized.

In comparing the two classifiers, recall that the fuzzy classifier uses a conjunction of membership functions that are computed serially. If any one component has overlap between two SECFs, the confidence level will be lower, and distinguishing SECFs may be harder. The NN, on the other hand, uses all six components simultaneously, and thus has a better opportunity to distinguish SECFs. The NN difficulties stem from the fact that with very close or overlapping sets, backpropagation may find local rather than global minima, a well known problem [15]. This suggests that altering the training discipline may improve the situation.

C. Modified Neural Network Training

In an effort to improve training, several other methods were tried, learning rate adjustment, adaptive learning rates, and Nguyen-Widrow weight initialization [15]. None yielded significantly-improved results.

Ultimately, a two-step training process was adopted. First, the classifier is trained using a small set of prototype vectors. For each SECF, 10 prototype vectors are randomly generated to be within 1 standard deviation of the class mean. Figure 8 illustrates the well-defined separation that typically results. The weights and biases learned by the network using the prototype vectors are then used as starting values for the second phase of training, in which the complete training set is used.

TABLE II
NEURAL NET CLASSIFIER RESULTS

set	hidden	TSE	epochs	LR	good	misses	unclass.	%success
pentagon-11	10	9.5	766	0.1	1818	11	16	98.5
plug-7	3	12.0	13000	0.2	177	1	1	98.9
peg-11	8	103	600	0.1	1119	192	110	78.7
peg-9	8	152	65300	0.1	3277	53	254	91.4
square-17	8	212	200	0.1	6250	833	1368	74.0
square-8	8	39.6	13000	0.1	1965	16	3	99.0

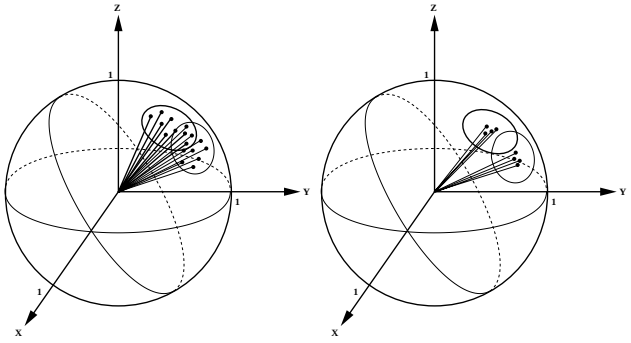

Fig. 8. Overlapping Clusters: Generating a Small Set of Prototype Vectors for the First Phase of Training

The new process yielded dramatically improved results for *peg-11*, *peg-9*, and *square-17* (see Table III).

D. Execution Time

Execution time was measured for the *square-17* set, the worst case time of the sets investigated as it contains the largest number of SECF classes. Using a 133 MHz pentium PC, the fuzzy classifier ran in 0.50 msec and the NN classifier ran in 0.31 msec. Thus, the algorithm is more than fast enough for real-time force-based systems. It is difficult to meaningfully compare these results with other reported times that solve somewhat different and less general problems using different hardware. However, given processor differences, these times are roughly comparable to Mosemann's extension to Hirai and Asada's classifier, which considered only static friction [7].

VI. DISCUSSION

One of the more interesting results occured with the non-rigid part (*plug-7*). Even with a small training set, the results were quite good, especially for the NN. Such flexible parts are difficult to model; consequently, a sensor-based approach may be especially useful.

While we achieved quite reasonable results, we did not prove that we could distinguish between SECFs in all cases. The same resultant vectors may occur with different CF's, as in multi-point contacts, or for different SECFs. Also, the classifiers can not distinguish where the contact is made in the environment. Using a wrist force sensor as the sole sensor contributed to this. The addition of position information, even coarse information, might alleviate some problems.

There are tradeoffs in selecting a classifier. The fuzzy classifier is easier to train, as only means and standard deviations are computed. For objects with adequate separation among SECF classes, e.g., *square-17* or *pentagon-11*, it performs quite well. However, as it uses 6 one-dimensional membership functions, it does not handle skewed alignments well. The NN classifier achieves higher performance in the more general case, but requires more time-consuming training. An attractive extension to the fuzzy classifier might be to generate multivariate membership functions from the covariance matrix, to handle the general situation and at the same time provide fast classifier training.

Alternative SECF cluster representations are also worth considering. E.g., use of spherical coordinates could reduce the dimensionality. One of the reviewers suggested using PCCs [3]. Demonstration-based data collection could be used as the basis for supervised PCC learning; PCCs provide a SECF approximation. PCC concepts, however, would have to be extended to include uncertainties, as well as generating confidence levels.

VII. CONCLUDING REMARKS

Key to event-based robot programming by demonstration is real-time SECF identification (i.e., of the force-based qualitative state). We have shown that by projecting force and moment vector sets onto unit spheres, SECF identification can be reduced to determining the cluster in which a projected vector lies. Two efficient classifiers were presented, one based on fuzzy sets, and the other on a neural network, both providing a confidence measure in the face of friction and sensor noise. Experiments were run for 6 data sets to compare performance in different situations. In all cases, one or more of the methods yielded a success rate of at least 94%, in most cases better than 98%.

The method has been used successfully to learn simple assembly skills via demonstration [5]. The strategy may also be useful in other applications, e.g., teler-

TABLE III
MODIFIED NN CLASSIFIER RESULTS

Set	hidden	TSE	epochs	LR	good	misses	unclass.	%success
peg-11	8	87	20100	0.025	1392	17	12	98.0
peg-9	8	147	5900	0.050	3388	95	101	94.5
square-17	8	203	1200	0.050	8263	91	97	97.8

obotics, where an operator must perform contact operations, particularly in environments with limited structure or challenges such as time delays.

REFERENCES

[1] R.S. Desai and R.A. Volz, "Identification and verification of termination conditions in fine motion in presence of sensor errors and geometric uncertainties," in *Proceedings of the 1989 IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, May 1989, pp. 800–807.

[2] K. Kosuge, T. Fukuda, and H. Asada, "Acquisition of human skills for robotic systems," in *Proceedings of the 1991 IEEE International Symposium on Intelligent Control*, Arlington, VA, Aug. 1991, pp. 469–474.

[3] S. Hirai and H. Asada, "Kinematics and statics of manipulation using the theory of polyhedral convex cones," *International Journal of Robotics Research*, vol. 12, no. 5, pp. 434–447, Oct. 1993.

[4] A.O. Farahat, B.S. Graves, and J.C. Trinkle, "Identifying contact formations in the presence of uncertainty," in *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pittsburgh, PA, Aug. 1995, vol. 3, pp. 59–64.

[5] M. Skubic, *Transferring assembly skills to robots: Learning force sensory patterns and skills from human demonstration*, Ph.D. thesis, Texas A&M University, College Station, TX, 1997.

[6] M. Skubic and R.A. Volz, "Learning force-based assembly skills from human demonstration for execution in unstructured environments," in *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[7] H. Mosemann, A. Raue, and F. Wahl, "Identification of assembly process states using polyhedral convex cones," in *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, Detroit, MI, May 1999, pp. 2756–2761.

[8] B. J. McCarragher and H. Asada, "Qualitative template matching using dynamic process models for state transition recognition of robotic assembly," *Journal of Dynamic Systems Measurement and Control– Transactions of the ASME*, vol. 115, no. 2, pp. 261–269, June 1993.

[9] P. Sikka and B.J. McCarragher, "Monitoring contact using clustering and discriminant functions," in *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, Minneapolis, MN, Apr. 1996, vol. 2, pp. 1351–1356.

[10] G.E. Hovland and B.J. McCarragher, "Hidden markov models as a process monitor in robotic assembly," *International Journal of Robotics Research*, vol. 17, no. 2, pp. 153–168, Feb. 1998.

[11] K. Hara and R. Yokogawa, "Recognition of state in peg-in-hole by fuzzy schema," *Journal of Advanced Automation Technology*, vol. 4, no. 3, pp. 134–139, 1992.

[12] E. Cervera, A. del Pobil, E. Marta, and M. Serna, "Perception-based learning for motion in contact in task planning," *Journal of Intelligent and Robotic Systems*, vol. 17, no. 3, pp. 283–308, Nov. 1996.

[13] H.-J. Zimmermann, *Fuzzy Set Theory and Its Applications, Second Edition*, Kluwer Academic Publishers, Boston, MA, 1991.

[14] H. Asada, "Representation and learning of nonlinear compliance using neural nets," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 6, pp. 863–867, Dec. 1993.

[15] L. Fausett, *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, NJ, 1994.