Proceedings of the ASME Dynamic Systems and Control Division, presented at the ASME IMECE, Atlanta, GA, Nov 1996, DSC-Vol. 58, pp. 451-458.

AUTOMATIC IDENTIFICATION OF REMOTE ENVIRONMENTS

Timothy M. Schulteis and Pierre E. Dupont
Boston University
Department of Aerospace and Mechanical Engineering
110 Cummington Street
Boston, Massachusetts, 02215, U.S.A.
(617) 353-9596, FAX: (617) 353-5866, tschulte@bu.edu, pierre@bu.edu

Paul A. Millman and Robert D. Howe
Harvard University
Division of Applied Sciences
29 Oxford Street
Cambridge, Massachusetts, 02138, U.S.A.
(617) 496-9098, FAX: (617) 495-9837, millman@hrl.harvard.edu, howe@hrl.harvard.edu

ABSTRACT

A system that can process sensory information collected during telemanipulation tasks in order to automatically identify properties of the remote environment has many potential applications. These include generating model-based simulations for training operators in critical procedures and improving real-time performance when time delays are large. The research issues involved in developing such an identification system are explored, focusing on properties that can be identified from remote manipulator motion and force data. As a case study, a simple block-stacking task performed with a teleoperated two-fingered planar hand is considered. An algorithm is presented which automatically segments the data collected during the task, given only a general description of the temporal sequence of task events. Using the segmented data, the algorithm then successfully estimates the weight, width, height, and coefficient of friction of the two blocks handled during the task. This case study highlights the broader research issues which must be addressed in automatic property identification.

INTRODUCTION

Traditional teleoperated manipulation relies entirely on the human operator for perception and control. The operator perceives the remote environment visually and kinesthetically and then generates the appropriate commands to accomplish the task at hand. However, in many teleoperation applications, machine perception of the remote environment can also play an important role. For example, in remediation of toxic waste dumps, quantitative measurements of the size and weight of the containers helps to infer their contents and to determine optimal handling strategies (Griebenow, 1994). Similar considerations apply to collecting rock samples on interplanetary missions and to transporting and defusing explosives. Machine perception can also be used to develop models of the environment for use as real-time simulations. Simulations such as these have already been used to overcome stability problems related to teleoperation with significant time delays (Noyes and Sheridan, 1984; Bejczy, Kim, and Venema, 1990; Funda and Paul, 1991). They could also be used to train operators prior to critical operations.

The problem of automatic identification of remote object properties has received little attention in telemanipulation research. (In the one study we are aware of, Fyler (1981) demonstrated the ability to build up a picture of the shape of a remote object using a touch probe on the end of the robot manipulator arm.) By "automatic" we mean that the identification procedure is performed by the telemanipulation system with little or no input from the human operator. Ideally, such a system would be able to identify all parameters of interest by observing normal telemanipulation procedures, making the system transparent to the operator. In actuality, input may be required from operators in the form of information (e.g. task context) or special exploratory motions.

The environment identification problem is actually composed of three main sub-problems: task decomposition, data segmentation, and parameter estimation. It is assumed for this paper that the operator specifies not only the task to be performed but also the associated subtasks and states (task decomposition). Data segmentation is the reconstruction of the events or stages of a manipulation task based on streams of sensory data. This is important because the context or state of the system at any given time will dictate which parameters can be estimated at that time. For example, to measure the weight of a grasped object, the estimator must determine when it is freely supported by the manipulator, and to find its coefficient of friction with the manipulator, the estimator must determine when it is slipping against the gripper fingers. Once the data is segmented in time, parameter estimation can proceed using sensory data from within each of the segments. Individually, the manip-

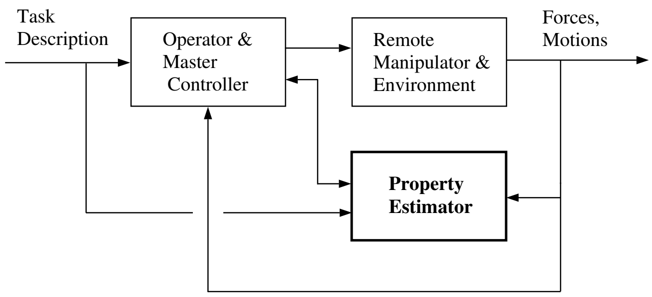

FIGURE 1: Block diagram of the property identification problem.

ulation data segmentation and mechanical parameter estimation problems have been studied extensively in other contexts, however, the application of these techniques to remote environment identification has not previously been pursued.

Segmentation of manipulation data has received considerable attention for skill transference from humans to autonomous robots (Pook and Ballard, 1993; Kang and Ikeuchi, 1994; Yang, Xu, and Chen, 1994). In those studies, the goal was to find the commands which resulted in successful execution of each subtask, which may not be necessary for property estimation. Other work has been directed at determining the manipulation strategies of humans (McCarragher, 1994) or evaluating the performance of teleoperator systems (Hannaford et al., 1991; Hannaford and Lee, 1991), where task decomposition can be used to relate performance to subtask attributes. For property estimation, segmentation may not require classification of every portion of the data stream.

Estimation of parameters from data in the appropriate time segments is a classic estimation problem (Ljung, 1987). Many of the parameters of interest are simple constants (e.g. mass, stiffness) and will be straightforward to estimate. In some applications, however, metal-on-metal contact between the gripper and grasped objects will generate noisy, rapidly changing force sensor signals, so more sophisticated algorithms may be required to determine those parameters that are conveyed by force information.

Figure 1 shows the flow of information between various components of the environment identification system. As the operator uses the teleoperated manipulator to execute tasks, the property estimator system monitors feedback from the manipulator, which in our case includes only force and motion signals. Properties that can be determined from these signals include: object geometry, mass, mass distribution, friction coefficients, stiffness, and surface roughness and waviness.

The remainder of this paper proceeds with a description of the experimental hardware, including our two-fingered, planar teleoperated hand. Next we perform post-facto segmentation of the force and motion data followed by identification of task-related object properties.

TELEOPERATED DEXTEROUS HAND SYSTEM

These experiments use a planar, two-fingered teleoperated hand system with finger tip force feedback (Howe, 1992). This system trades a limitation on the number of joints for a clean and simple mechanical design. The system has high bandwidth and large dynamic range, which permits accurate control of contact forces and small motions. The system is designed to execute tasks that humans usually accomplish with a precision pinch grasp between the thumb and index finger. For most tasks, the operator's wrist rests on the table top and the operator makes contact with the master only at the tips of the fingers (Fig. 2). The master and remote manipulators are kinematically identical, with two degrees of freedom in each finger, so finger tip position or force can be controlled within the vertical plane. The workspace is roughly circular and 75 mm in diameter. Parallelogram linkages maintain a constant vertical orientation of the finger tips, which precludes inappropriate torques on the operator's finger tips as the joints rotate. Two-axis strain gauge force sensors measure finger tip forces on both master and remote manipulator hands.

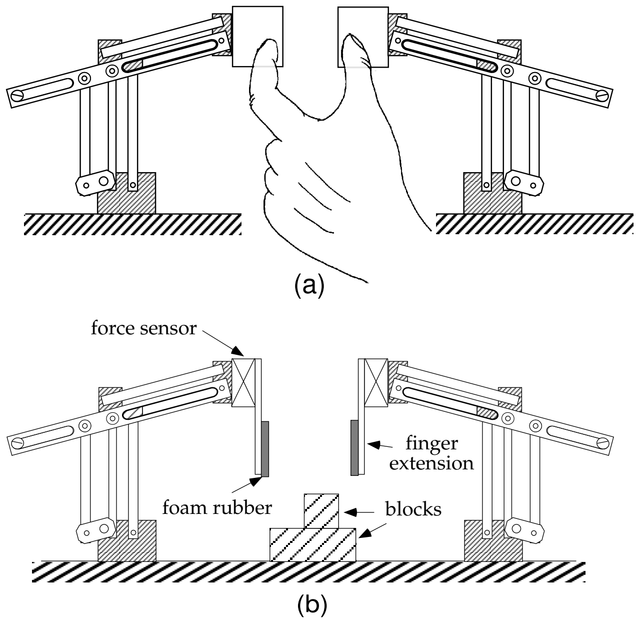

FIGURE 2: (a) Master manipulator with operator's hand. (b) Remote manipulator.

The controller uses a conventional bilateral force reflection control scheme. The joint angles of the master manipulator are the command inputs for position control of the joints of the remote manipulator, and the forces measured at the remote manipulator finger tips are the command inputs for force control of the master. Each finger is capable of applying a continuous tip force of at least 4 N. Flat, thin finger tips extending downward are mounted on the two remote manipulator fingers to facilitate manipulation of the rectangular blocks used in the experiments (Fig. 2b). The manipulator finger tips were covered with a 2 mm layer of closed-cell foam rubber to increase compliance and friction.

BLOCK-STACKING TASK

Pick-and-place tasks are a convenient starting point for the study of automatic identification techniques, because the grasp-

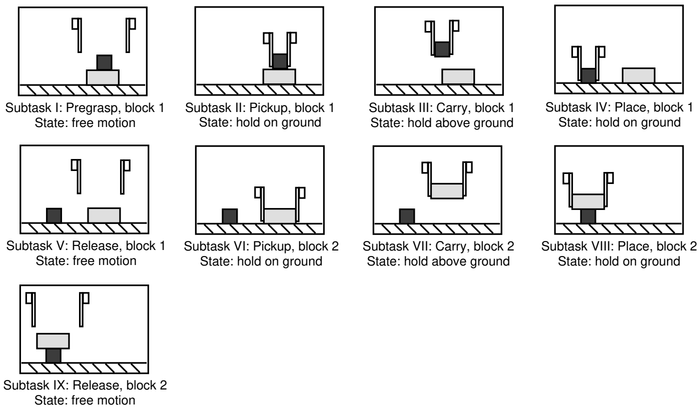

FIGURE 3: Subtasks of the stacking task.

ing and lifting actions that comprise these tasks are an essential part of many telemanipulation operations. These tasks are also amenable to automatic identification, as the parameters of interest and task segments are readily defined. To include interactions between objects in the remote environment, here we analyze a block stacking task, which requires reversing the positions of two aluminum blocks, stacked one atop the other, using the teleoperated hand system. The operator moves the top block off of the stack and onto the ground, and then places the other block on top of it. Figure 3 shows the progression of the task through the various subtasks.

The trials for this case study were performed by one of the authors after practicing sufficiently to become proficient at the task. Six sensor signals were recorded for each of the two remote manipulator fingers: two joint angles, two joint velocities, and horizontal and vertical components of finger tip force. The signals were collected at 50 ms intervals during the course of the task, for a total of 10 seconds. The forward kinematic relations permitted calculation of endpoint position and velocity of each finger, as shown in the subsequent figures. In the following sections, we investigate the segmentation and parameter estimation processes using this data.

SEGMENTATION

To relate the force and displacement data streams to the parameters of interest, we divide the task into *subtasks*, the sequence of operations required to execute the task, and relate these to *system states*, the description of the physical configuration of the remote manipulator and environment. This approach is an extension of the syntactic method used by Hannaford et al., (1989) for segmentation of peg-in-hole telemanipulation tasks. As illustrated in Fig. 3, the example task begins with subtask I, *pregrasp of block 1*, followed by subtasks *pickup*, *carry*, *place*, and *release* of the first block. The ensuing subtasks for the second block are *pickup*, *carry*, *place*, and *release*.

Each subtask in Fig. 3 is also labeled with a *system state*, which describes the configuration of the manipulator robot and environment at a specific time, including contact and constraint conditions. The physical situation described by the state then implies which parameters can be identified in that subtask. Three states are shown: *free-motion* of the fingers, *hold-on-ground*, and *hold-above-ground*. A fourth possible state, *hold-on-ground-with-sliding*, is not shown in Fig. 3, although it could occur during any *hold-on-ground* state. Each state can describe more than one subtask, as physical configurations are repeated as the task progresses. Although it will not necessarily be true for more complex tasks, in this initial study we assume that states can be determined by local values of motion and force signals, without reference to the history of the task. Thus the *free-motion* state can be distinguished by the absence of finger tip forces.

In this example, we begin the parameter identification procedure after the task is complete, so that the entire position, velocity, and force records are available. We also specify the expected sequence of subtasks and their corresponding states for each task. The correspondence between states and the parameters which can be identified in that state is also given explicitly, although this information can be compiled in a database for use with a variety of tasks; for instance, the weight of a grasped object can be determined in every hold-above-ground state. The operator must then relate the specific parameters which are to be identified to the generic parameters, so that the system can, for example, distinguish the weights of blocks 1 and 2.

The automatic segmentation algorithm uses Boolean combinations of thresholded force and velocity data to identify the system state at each time. Once the system state at each time is determined, the subtasks are identified using a priori knowledge of the expected sequence of subtasks, and additional historical information. For example, carrying block 1 and carrying block 2 both occurred during a hold-above-ground state; however, they were separated by a free-motion state, a fact used to distinguish the two subtasks.

Table 1 shows the sensor signals required for estimating the environment parameters during the first four subtasks. The system state for each subtask is also listed. Most of the parameters pertain to the first block (with the exception of the height of the lower block). Properties of the second block (and the height of the first block) can be estimated during the last five subtasks.

Automatic Identification of Subtasks and States

Given the property-subtask correspondence summarized in Table 1, the automatic identification of task states proceeded in three stages, similar to the qualitative reasoning approach that has been applied to the analysis of sensor signals from human task performance (McCarragher, 1994). In the first stage, the measured motions and forces at the finger tips were transformed into task coordinates and thresholded. The resulting threshold functions were tri-valued, that is, they were assigned values of "+", "0", or "-". In the second stage, the thresholded data were combined using Boolean operators to assign state labels. Once the states were identified, additional processing identified the subtasks. Here we describe the details of the state and subtask identification procedures.

Because these experiments used a multifingered hand, the finger tip motion and force data must be transformed to find the net motion and force of the grasped object in task space. This requires computation of averages and differences of the horizon-

TABLE 1: Relevant Signals and System States for Estimation of Object Parameters During the First Four Subtasks

		Object Parameter			
Subtasks	System State	Weight Block #1	Width Block #1	Height Block #2	Friction of Fingers Against Block #1
Pregrasp, Block #1	free-motion	-	-	-	-
Pickup, Block #1	hold-on-ground	-	-	vertical positions	horiz. and vertical forces
Carry, Block #1	hold-above-ground	vertical forces	horizontal positions	-	horizontal and vertical forces
Place Block #1	hold-on-ground	-	-	vertical positions	horizontal and vertical forces

tal (x) and vertical (y) components of the two finger motions. The average motion of the two fingers is a measure of rigid body motion, and the difference in finger tip motions corresponds to gripping motion. Horizontal and vertical components of the finger tip forces were similarly transformed. The sum of the forces from the two fingers was the net force applied to the environment (including the block and the ground), and grip force was the minimum horizontal component of the two (opposing) finger tip forces. Because each finger of our system has only two degrees of freedom and flat finger tips, and because the objects were flat-sided blocks, the kinematic transformations of endpoint forces and motions to task space was trivial. In general, these transformations for multifingered hands require the use of robotic grasp analysis (Kerr and Roth 1986).

Once the task motions and forces were obtained, the velocity and force data were passed through thresholding filters with equal positive and negative thresholds. The velocity thresholds were ± 5 cm/sec, and the force thresholds were ± 0.05 N. Position and force data for one trial of the stacking task are shown below in Figs. 4 and 5. The progression of the task can be discerned in these figures. The side to side pattern of motion can be seen in Fig. 4a, up and down motion in Fig. 4b, and grasp and release in Figure 4c. The vertical offset of the two fingers (Fig. 4d) is not of interest in this task, but has been included for completeness. The forces applied to the environment are shown in Fig. 5. Horizontal forces (Fig. 5a) are generally zero during the carry and free-motion subtasks, and non-zero during pickup and placement subtasks. Net vertical force (Fig. 5b) is also zero during free-motion, but positive during carry subtasks, and negative or zero during pickup and placement. The holding subtasks are clearly visible in Fig. 5c as large, positive grip forces.

These correlations between sensor data and task states were formalized into rules for automatically identifying the various states, as summarized in Table 2. Free-motion was defined as any time that both components of the force on both finger tips were zero. The hold-on-ground state was active whenever the grasp force was positive (i.e., greater than the positive threshold) and the total vertical force exerted by the fingers was not upward (i.e., negative or zero) and the average vertical velocity of the fingers was near zero. The sliding-on-ground state is the same as hold-on-ground, with the added condition that the average horizontal velocity was non-zero. Hold-above-ground was active when the grasp force was positive and the sum of the vertical forces was upward (positive). Note that this criterion for the hold-above-ground state can include a brief period just before and after the block lifts, when the vertical force is positive but the block is still in contact with the ground. This is unavoidable if the state identification algorithm uses only instantaneous data without reference to the prior state of the system. This approach proved adequate for this simple task.

Data for several trials of the block-stacking task were processed using the automatic state identification procedure. The segmentation algorithm divided the data into sections which corresponded closely to those selected by hand. For these data sets, the results of thresholding were not especially sensitive to the threshold values, although selecting appropriate thresholds can be difficult when noisy manipulators and a larger range of tasks and operators are involved (Hannaford and Lee, 1991).

Figure 6 shows the results of applying the procedure to the data in the preceding figures. The value of the plotted function denotes the identified state of the system at that time. A value of 1 corresponds to free-motion, a value of 2 to hold-above-ground, and a value of 3 to hold-on-ground. Times at which sliding on the ground was identified are labeled with an "x". A value of zero indicates that none of the four possible states was identified. Accordingly, data from these times were not used in the identification of any object properties. These "dropouts" are caused mainly by transients in the signals just after lift-off or impact of the blocks. While these unidentified points are not desirable, they are not a problem as long as at least some data samples are successfully identified for each subtask. Using this convention, the expected (and observed) progression of states for this task was: 1, 3, 2, 3, 1, 3, 2, 3, 1. Subtasks I-IX were assigned to the data stream from the state information using the list of the anticipated order of the states.

A weakness of the simple segmentation algorithm used for this study is that it could potentially oscillate between the hold-on-ground state and the hold-above-ground state if the manipulated objects did not make and break contact cleanly. Although the data collected for this example did not exhibit this compli-

TABLE 2: Qualitative Values of Sensed Parameters Which Define Task States

	Sensed Parameters					
State	Vx_ avg	Vy_ avg	Fx_ avg	Fy_ avg	Fx_ grip	Fy_ shear
free-motion			0	0	0	0
hold-on-ground		0		- or 0	+	
hold-on-ground, sliding	+ or -	0		- or 0	+	
hold-above-ground				+	+	

TABLE 3: Methods of Property Estimation

Property	*Subtask*	*System State*	*Formula for Estimate*
Weight (Block 1)	Subtask III Carry Block 1	hold-above-ground	$W = \Sigma \frac{F_{y1}+F_{y2}}{N_{III}}$ (Average sum of vertical forces for finger1 and finger2)
Height (Block 2) Part a:	Subtask II Pick up Block 1	hold-on-ground	$y_{II} = \Sigma \frac{y_2}{N_{III}}$ (Average vertical position of right finger)
Part b:	Subtask IV Place Block 1	hold-on-ground	$y_{IV} = \Sigma \frac{y_2}{N_{IV}}$ (Average vertical position of right finger)
Final:			$h = \lvert y_{II} - y_{IV} \rvert$ (Absolute value of the difference in average height)
Width (Block 1)	Subtask III Carry Block 1	hold-above-ground	$w = \Sigma \frac{x_2 - x_1}{N_{III}}$ (average difference in horizontal position)
μ (Block 1)	Subtasks II-IV Pick up Block 1, Carry Block 1, and Place Block 1	hold-on-ground and hold-above-ground	$\mu \geq max \lvert \frac{F_{y1}}{F_{x1}} \rvert$ (Max. absolute value of vertical force over horizontal force)

TABLE 4: Comparison of Estimated (mean and standard deviation) and Actual Environment Parameters

	Block 1		*Block 2*	
Property	*Estimated Value*	*Actual Value*	*Estimated Value*	*Actual Value*
Weight	0.352 ± 0.024 N	0.343 ± 0.0005 N	0.425 N ± 0.071 N	0.437 ± 0.0005 N
Height	1.75 ± 0.01 cm	1.60 ± 0.01 cm	1.29 ± 0.02 cm	1.27 ± 0.01 cm
Width	1.47 ± 0.01 cm	1.59 ± 0.01 cm	2.36 ± 0.01 cm	2.54 ± 0.01 cm
Lower Bound on μ	0.49	-	1.06	-

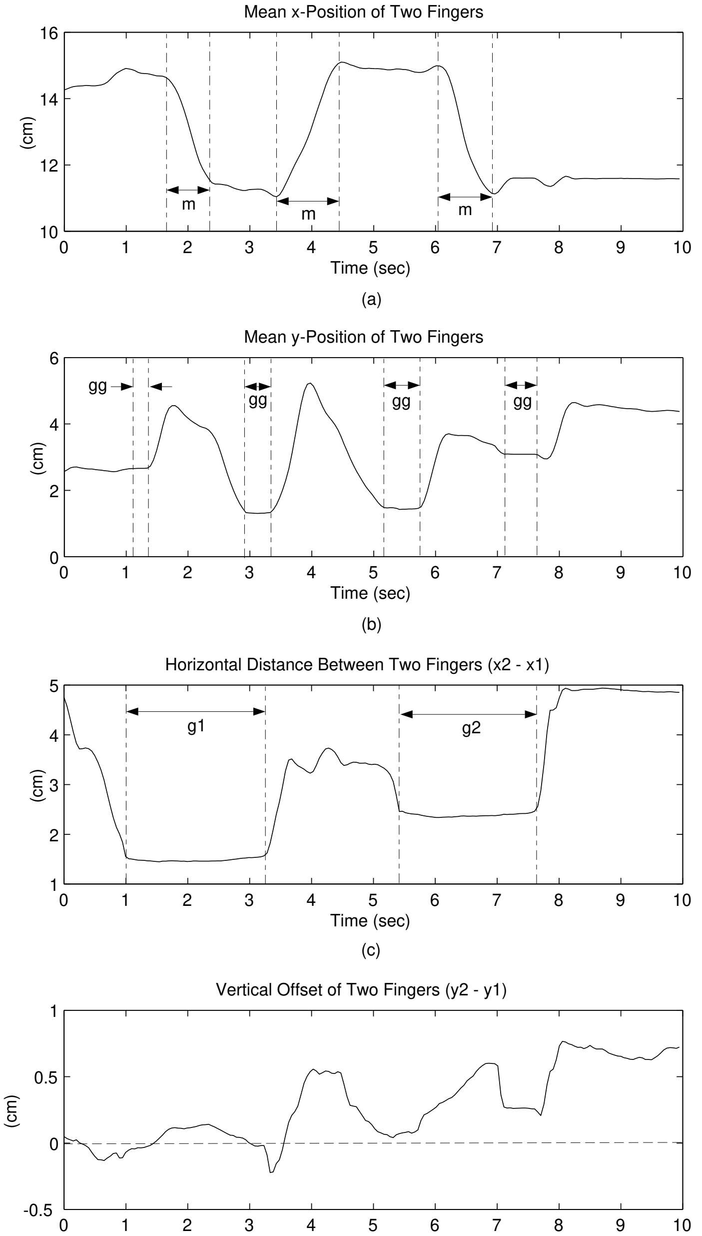

FIGURE 4: Horizontal and vertical positions of the two fingers. Legend: m = Movement of fingers to a new position; gg = Grasping an object on the ground; g1 = Grasping object 1; g2 = grasping object 2.

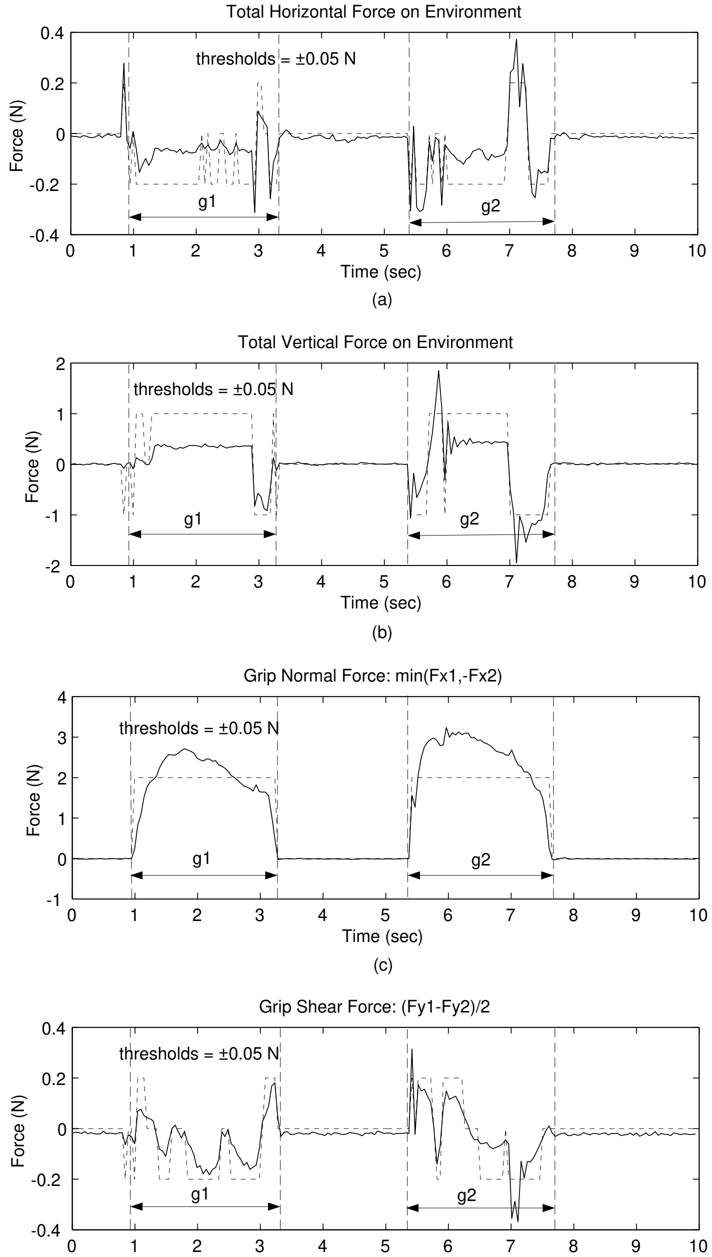

FIGURE 5: Forces: (a) Total horizontal force on environment; (b) Total vertical force on environment; (c) Grip force; (d) Vertical shear force. Legend: g1 = Grasping object 1; g2 = Grasping object 2.

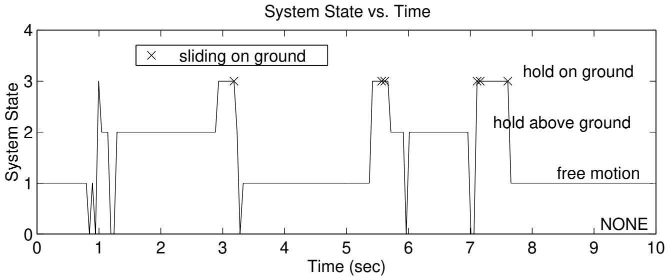

FIGURE 6: Result of automatic system state identification procedure.

cation, future segmentation algorithms should be robust to this problem.

IDENTIFICATION OF OBJECT PROPERTIES

For the sake of brevity, only four illustrative properties were estimated from the data, namely block weight, height, width, and μ, the coefficient of static friction between the block and the fingers of the robot hand. Table 3 lists each property along with the subtasks in which the appropriate measurements can be collected. The last column of the table contains the simple formulas used to estimate the object properties. Because our goal is to outline the steps of the identification process and illustrate the key issues, no attempt has been made to find an optimal estimator. This table describes object property estimation during manipulation of the first block (Subtasks II-IV); for estimation of the analogous properties during manipulation of the second block (Subtasks IV-VIII), each instance of "block 1" in the table is simply replaced by "block 2" and vice versa.

Ideally, the weight of block 1 was the sum of the vertical forces measured at each finger tip during the "carry block 1" subtask. The estimate was the average sum over all samples in the subtask (i.e. over N_{III} , the number of samples in subtask III). The weight estimate is sensitive to segmentation boundaries, due to noise during the transition period between subtasks. Therefore, the weight estimate arbitrarily excluded force data during the first and last 5% of the carry subtask.

The width of each block is simply the average horizontal distance between the finger tips over the entire carry subtask. In contrast, the height estimate uses interactions between objects in the remote environment to determine the desired parameters. This requires multiple estimation steps because it is formed from two discrete measurements. For example, the height of block 2 is found from the difference between the vertical position of the fingers when block 1 is lifted from atop block2 and when it (block 1) is subsequently placed on the table. Similarly, moving block 2 gives the height of block 1.

Note that for these experiments, we assume static friction can be modeled as a single constant μ. Ideally, μ would be measured at the onset of slip, but since slippage of the blocks between the fingers was not detectable, only a lower bound of μ was obtained. If slip did occur, the maximum value of μ recorded should correspond to the coefficient of static friction, recorded at the onset of slip. In many telemanipulation tasks operators expressly avoid slips, so accurate determination of μ could require a special test procedure.

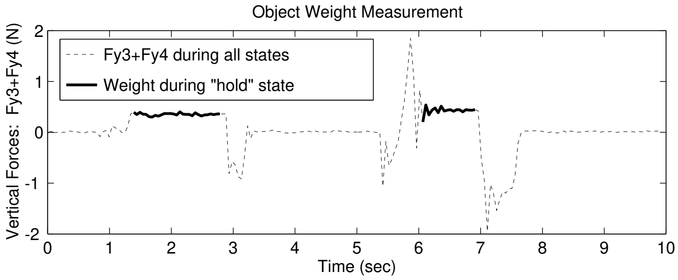

FIGURE 7: Object weights. The solid portions are the segments of data used for weight estimation.

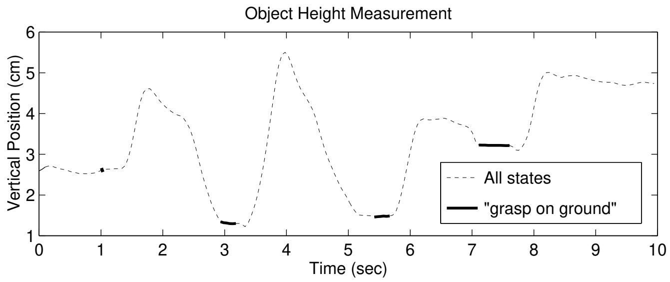

FIGURE 8: Object Heights. The solid portions are the segments of data used for height estimation. Object 2 height is the difference between the first and second solid lines, while object 1 height is the difference between the third and fourth solid lines.

RESULTS

The estimated parameters for one trial of the block-stacking task are listed in Table 4, along with actual values, as measured by a laboratory balance and calipers. Applying the algorithm to a different trial of the block stacking task gave similar estimates of the parameters. Certain trials exhibited significant vibrations following state transitions suggesting the need for data filtering prior to segmentation.

Figures 7 - 8 are plots of the measurements used to estimate object weight and height. The portions of the data that were used in the estimates are indicated by solid lines. Figure 7 shows the force measurements used to find the average weight during the hold subtask. Note that there are two intervals of measurements, indicating that two objects were held. The large oscillations that occur during the second interval are due to inertial forces and the object impacting the surface. The average estimates of the weights are within 3% of the actual values. The standard deviation is relatively large for block 2 due to the oscillations in the force signals.

The first two solid lines in Figure 8 show the heights during the two hold-on-ground states, when block 1 is moved from the top of block 2 to the ground. The difference in these heights gives the height of the block originally beneath it. Likewise, the last set of two solid lines represents grasping and moving block 2 from the ground to the top of block 1. Notice that the intervals for these subtasks are fairly short, but even though there are only a few points in the interval, the estimated heights of blocks 1 and 2 are within 10% and 2% of their respective measured dimensions. The estimated height of block 1 shows greater error, perhaps due to slipping of the block within the fingers at the instant when the block makes contact with the ground. The errors in the estimates of the widths of the blocks were approximately 7%.

CONCLUSIONS

Automatic identification of the properties of remote environments promises to improve the handling of hazardous materials, and to aid in the construction of environment models for stabilizing telemanipulation with time delays. Through the analysis of a simple block stacking task, we have demonstrated that there are three principal functions in automatic identification: task decomposition, data segmentation, and parameter estimation. In this initial study, we started with an operator-specified task description that divided the task into a sequence of subtasks, with associated system states and identifiable parameters. Segmentation and estimation algorithms then used force and motion signals to determine several basic properties of the blocks, including weight, size, and a lower bound on the coefficient of friction between the blocks and the manipulator finger tips. Key issues we hope to address in future work include development of robust segmentation and estimation techniques for a variety of manipulators and tasks, and determining the appropriate role of user interaction in remote parameter identification.

REFERENCES

Bejczy, A. K., Kim, W. S., and Venema, S. C., 1990, "The Phantom Robot: Predictive Displays for Teleoperation with Time Delay," *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, Vol. 1, May, pp. 546-551.

Funda, J., and Paul, R. P., 1991, "Efficient Control of a Robotic System for Time-Delayed Environments," *91 ICAR. Fifth International Conference on Advanced Robotics. Robots in Unstructured Environments.*, Vol. 1, June, pp. 219-224.

Fyler, D., 1981, "Computer Graphic Representation of Remote Environments Using Position Tactile Sensors," SM thesis, MIT, August.

Griebenow, B.E., 1994, "Buried-waste Integrated Demonstration Retrieval Projects." Annual Meeting of American Nuclear Society, New Orleans, LA, 19-23 June, *Trans. Am. Nucl. Soc.*, Vol. 70, p. 402.

Hannaford, B., and Lee, P., 1991,"Hidden Markov Modal Analysis of Force/Torque Information in Telemanipulation," *International Journal of Robotics Research*, Vol. 10, No. 5, October, pp. 528-539.

Hannaford, B., Wood, L., Guggisberg, B., McAffee, D., and Zak, H., 1989, "Performance Evaluation of a Six-Axis Generalized Force-Reflecting Teleoperator," JPL Publication 89-18, Pasadena, CA.

Hannaford, B., Wood, L., McAffee, D. A., and Zak, H., 1991, "Performance Evaluation of a Six-Axis Generalized Force-Reflecting Teleoperator," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 21, No. 3, May/June, pp. 620-633.

Howe, R. D., 1992, "A Force-Reflecting Teleoperated Hand System for the Study of Tactile Sensing in Precision Manipulation," *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, Nice, France, Vol. 2., May, pp. 1321-1326.

Kang, S. B., and Ikeuchi, K., 1993, "Toward Automatic Robot Instruction from Perception – Recognizing a Grasp From Observation," *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 4, pp. 432-443.

Kerr, J., and Roth, B., 1986, "Analysis of Multifingered Hands," *International Journal of Robotics Research*, Vol. 4, No. 4, pp. 3-17.

Ljung, L., 1987, *System Identification: Theory for the User*, Prentice-Hall, Englewood Cliffs, NJ.

McCarragher, B. J., 1994, "Force Sensing from Human Demonstration Using a Hybrid Dynamical Model and Qualitative Reasoning," *Proceedings of the 1994 IEEE Conference on Robotics and Automation*, Vol. 1, May, pp. 557-563.

Noyes, M. V. and Sheridan, T. B. 1984. "A Novel Predictor for Telemanipulation through a Time Delay," *Proceedings of the Annual Conference on Manual Control*, Moffett Field, CA, NASA Ames Research Center.

Pook, P. K., and Ballard, D. H., 1993, "Recognizing Teleoperated Manipulations," *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 2, May, pp. 578-585.

Yang, J., Xu, Y., and Chen, C. S., 1994, "Hidden Markov Model Approach to Skill Learning and Its Application to Telerobotics," *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 5, October, pp. 621-631.