Vibrotactile Recognition and Categorization of Surfaces by a Humanoid Robot

Jivko Sinapov, *Student Member, IEEE*, Vladimir Sukhoy, Ritika Sahai, and Alexander Stoytchev, *Member, IEEE*

***Abstract*—This paper proposes a method for interactive surface recognition and surface categorization by a humanoid robot using a vibrotactile sensory modality. The robot was equipped with an artificial fingernail that had a built-in three-axis accelerometer. The robot interacted with 20 different surfaces by performing five different exploratory scratching behaviors on them. Surface recognition models were learned by coupling frequency-domain analysis of the vibrations detected by the accelerometer with machine learning algorithms, i.e., support vector machine (SVM) and *k*-nearest neighbor (*k*-NN). The results show that by applying several different scratching behaviors on a test surface, the robot can recognize surfaces better than with any single behavior alone. The robot was also able to estimate a measure of similarity between any two surfaces, which was used to construct a grounded hierarchical surface categorization.**

***Index Terms*—Behavior-Based Systems, Force and Tactile Sensing, Learning and Adaptive Systems, Recognition.**

I. Introduction

In humans, the sense of touch is fundamental for both detecting and learning the properties of everyday objects. For example, one only needs to slide his index finger across a novel object in order to recognize its texture. Not surprisingly, there has been a growing interest in developing sensors and algorithms that would enable a robot to use and interpret tactile feedback while manipulating objects [1].

Psychologists and neuroscientists have discovered two different sensory modalities that are used to encode surface properties: a *tactile* sensory modality for coarse surfaces and a *vibrotactile* sensory modality for fine surfaces [2]. The former involves specialized cortical neurons that detect spatial variations through slowly adapting SA1 mechanoreceptors in the skin [2]. The vibrotactile sensory modality, on the other hand, is facilitated by cutaneous vibrations detected via the Pacinian corpuscles mechanoreceptors [2], [3].

Other research has shown that humans explore the tactile properties of objects through the use of a number of behaviors, which are commonly referred to as *exploratory procedures* [4] or *exploratory behaviors* [5]. For example, scratching an object can inform us of its roughness, while lifting it can inform us of its weight. Research in developmental psychology has repeatedly shown that exploratory behaviors coupled with tactile feedback are fundamental for infants' object perception [6], [7].

Inspired by these findings from psychology, this paper proposes the use of an artificial fingernail with a built-in three-axis accelerometer sensor for vibrotactile perception of common household surfaces. The accelerometer sensor can measure vibrations in the finger as the robot scratches different surfaces. Following research from developmental psychology, our humanoid robot was programmed with exploratory scratching behaviors, which were used to recognize surfaces and to form a hierarchical surface categorization. To solve these tasks, frequency-domain analysis was applied on the accelerometer measurements in order to extract spectrotemporal features from each interaction. The Support Vector Machine and the k-Nearest Neighbors learning algorithms were used to learn a surface recognition model based on these features. Using the learned models, the robot was able to estimate the similarity between any two surfaces and to learn a hierarchical surface categorization grounded in its own experience with them.

Twenty different surfaces, which were made of various materials, were used in the experiments. The results show that the robot recognized surfaces with high degree of accuracy. The results also show that the use of multiple exploratory behaviors can be crucial for achieving good recognition performance.

II. Related Work

Findings in psychology have shown that the tactile sensory modality is necessary to capture many object properties (e.g., roughness, texture, etc.) [8], [9]. More specifically, psychologists and neuroscientists have demonstrated that certain receptors in the skin are capable of detecting minute vibrations as the finger slides across a surface, thus enabling discrimination between fine textures [2], [3]. According to Lederman and Klatzky [4], tactile object exploration is facilitated by *exploratory procedures*. For example, to detect the roughness of a surface, a person might slide his finger across it; to detect its temperature, a person might touch it, etc. [10]. Studies have also shown that tactile exploratory behaviors are commonly used by infants when exploring a novel object [6]. For example, Stack and Tsonis [7] have reported that, in the absence of visual cues, 7-month-old infants use more efficient tactile exploratory strategies and can perform tactile surface recognition to some extent.

The importance of the sense of touch for biological organisms has lead to an increased interest in tactile sensors and their applications in robotics. For example, the goal of the ROBOSKIN project, which was recently funded by the European Commission, is to develop novel touch sensors for

Manuscript received July 31, 2010; revised December 23, 2010; accepted March 8, 2011. This paper was recommended for publication by Associate Editor M. Valle and Editor W. K. Chung upon evaluation of the reviewers comments. This paper was presented in part at a workshop held at the Ninth IEEE Robotics and Automation Society International Conference on Humanoid Robots, Paris, France, Dec. 7, 2009. The authors are with the Developmental Robotics Laboratory, Iowa State University, Ames, IA 50011 USA (e-mail: jsinapov@iastate.edu; sukhoy@iastate.edu; sahai@iastate.edu; alexs@iastate.edu).

an artificial skin that can cover large patches of the robot's body [11], [12]. The robotic skin is designed with flexible and modular components that can be easily reconfigured to the body morphology of a new robot. An early prototype of the skin has already been installed on the iCub robot. Another goal of the project is to use the skin sensor during social learning tasks, in which a human provides corrective feedback by touching the robot's hand to indicate a desired movement direction [13], [14].

Other research has focused on developing tactile-sensing technologies for robotic fingers [15], [16], [17], [18], [19], [20], [21], [22]. For example, Howe and Cutkosky [16] have developed a robotic finger with an artificial rubber skin, equipped with a piezoelectric polymer transducer that measures the changes in pressure induced as the sensor slides over a surface. It was shown that minute features (as small as 6.5 μm) could be detected on surfaces by sliding the sensor across them. Computer vision methods for surface perception have been explored as well [23].

Tanaka *et al.* [15] developed an artificial finger that uses strain gauges and polyvinylidene fluoride (PVDF) foil to generate tactile feedback when sliding across a surface. In subsequent experiments, they demonstrated how their sensor can detect roughness and temperature changes in the textures of six different fabrics [24]. A similar sensor was developed by Hosoda *et al.* [17]. By applying two different exploratory behaviors – *pushing* and *rubbing* – their robot was able to distinguish between five different materials. A robotic finger with randomly distributed strain gauges and PVDF films was also proposed by Jamali and Sammut [22]. In their experiments, a Naive Bayes classifier coupled with the Fourier coefficients of the sensor's output was used to recognize eight different surface textures.

Three-axis force sensors have also been used for tactile perception. For example, Beccai *et al.* [25] used a 3-axis microelectromechanical-system (MEMS) sensor to perform slip detection. A similar sensor was also used by de Boissieu *et al.* [26] to capture the high-frequency vibrations that occur when rubbing a surface. In that study, the force sensor was mounted on a plotter printer and was able to distinguish between ten different paper surfaces with reasonable accuracy (approximately 61%).

In another line of research, inexpensive accelerometers have been proposed by Romano *et al.* [27], [28] for the purposes of recording and reproducing tactile sensations. Howe and Cutkosky [29] have also developed a sensor that can detect tactile vibrations using a three-axis accelerometer, providing feedback that was useful for detecting if an object has moved after being grasped (i.e., slip detection). They estimated that the sensor's output was most dependent on the sliding speed, somewhat dependent on the surface roughness, and least dependent on the applied normal force [29].

The sensor presented in this paper uses a similar three-axis accelerometer to capture vibrotactile feedback and was previously introduced by Sukhoy *et al.* [30]. In contrast with previous work (including [30]), the humanoid robot described here performed both *surface recognition* and *surface categorization*. For the first task, the robot was evaluated on how well it could recognize the identity of a particular surface, given vibrotactile sensory feedback. For the second task – surface categorization – the robot was tasked with 1) grouping surfaces into category types based on its sensorimotor experience with them and 2) recognizing the category memberships of novel surfaces. To solve these problems, the robot used five exploratory scratching behaviors, which it applied to the different surfaces that it explored. Twenty different surfaces were used in the experiments, which is one of the largest number of surfaces reported in the robotics literature to date. The next section describes the artificial fingernail sensor and the experimental setup.

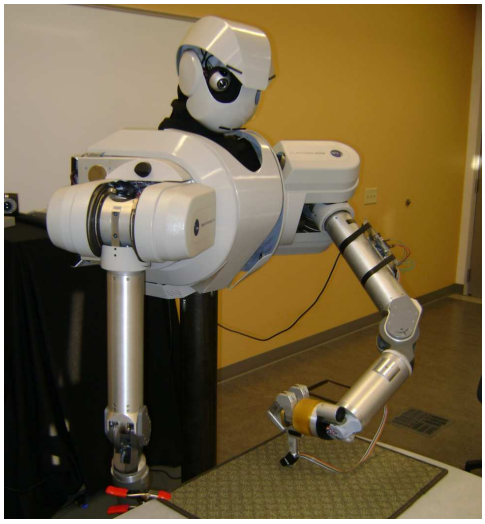


Fig. 1. Humanoid robot used in the experiments, which is shown here performing one of five scratching behaviors on a test surface.

III. Experimental Setup

A. Robot

The robot used in the experiments was an upper-torso humanoid robot with two Barrett whole arm manipulators (WAMs) for arms. Each WAM was equipped with the three-finger Barrett hand (BH8-262). Fig. 1 shows the robot performing an exploratory scratching behavior on one of the 20 surfaces used in the experiments.

B. Vibrotactile Sensor

The robot's vibrotactile sensor consisted of an artificial fingernail made of ABS plastic and the ADXL345 3-axis digital accelerometer mounted on the EVAL-ADXL345Z evaluation board. Both the accelerometer and the evaluation board were manufactured by Analog Devices. The accelerometer's output rate was 400.0 Hz using ten-bit resolution with a range of ± 2 g for each axis. The ADXL-345 accelerometer uses an on-board digital low-pass filter but does not have any analog anti-aliasing filters.

The ABS plastic fingernail was designed with Computer-aided design software and printed using a rapid prototyping 3-D printer. The sensor, along with its dimensions, is shown in Fig. 2. The EVAL-ADXL345Z accelerometer evaluation board was mounted on the fingernail, which, in turn, was attached to

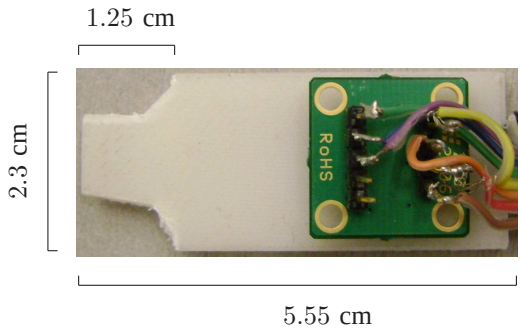


Fig. 2. The artificial fingernail with the three-axis accelerometer sensor. The thickness of the fingernail was 0.3175 cm (1/8th of an inch).

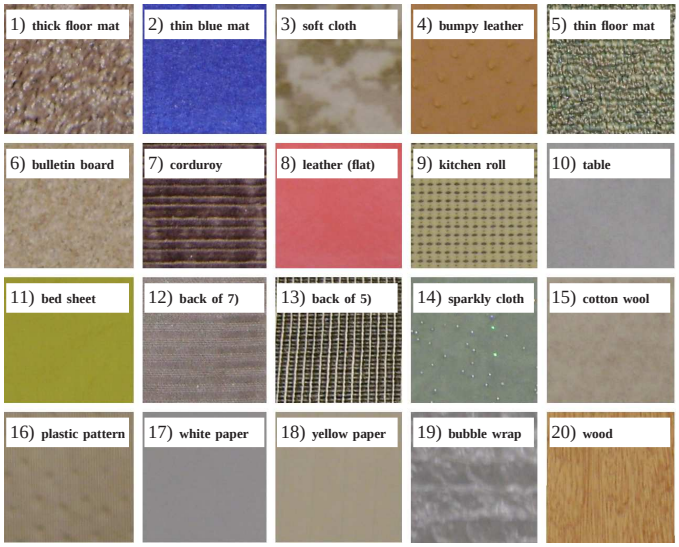


Fig. 3. Twenty surfaces scratched by the robot. 1) Thick floor mat; 2) thin blue mat; 3) soft cloth; 4) leather with bumps; 5) thin floor mat; 6) bulletin board:; 7) corduroy; 8) leather (flat); 9) plastic kitchen roll; 10) table; 11) bed sheet; 12) back of corduroy; 13) back of thin floor mat; 14) cloth with sparkles; 15) cotton wool (back of 8); 16) plastic pattern (back of 4); 17) paper, white; 18) paper, yellow; 19) bubble wrap; 20) wood. In addition, a 21st "surface" was added to the dataset as a control condition, which corresponded to the robot scratching in mid air.

the middle finger (i.e., F3) of the robot's left hand such that its tip protruded from the robot's finger. The sensor was attached to the finger with several layers of electrical tape, which was sturdy enough to prevent it from sliding (see Fig. 4). When the robot performed a scratching behavior, the vibrations of the fingernail were captured by the attached accelerometer. The accelerometer data were transferred to the PC over a universal serial bus (USB) at 400 Hz using the Arduino Duemilanove microcontroller. The sampling-frequency limitation was due to the limited serial port bandwidth of the Arduino board that was used to communicate with the accelerometer.

C. Surfaces

The robot performed exploratory scratching behaviors on the 20 different surfaces shown in Fig. 3. The surfaces were made of materials such as cloth, leather, wood, rubber, paper, and plastic. Some of the surfaces were specifically chosen to be similar to each other and, thus, were hard to distinguish,

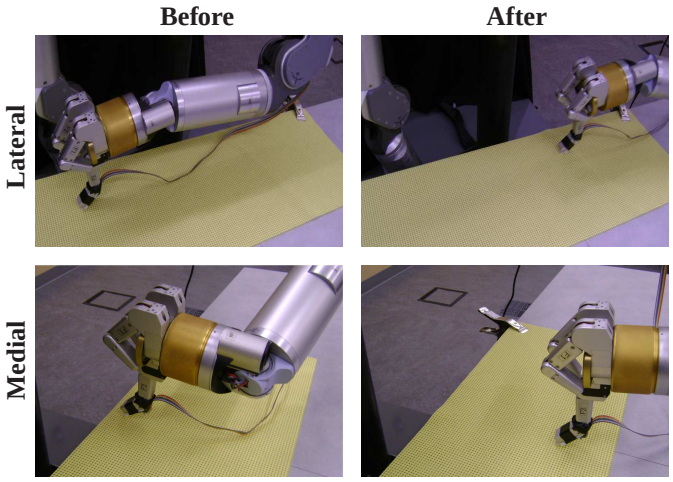


Fig. 4. Before and after images for two of the scratching trajectories performed by the robot on the *plastic kitchen roll* (i.e., surface 9).

TABLE I
THE FIVE EXPLORATORY SCRATCHING BEHAVIORS

Behavior	Sliding Direction	Duration
lateral-fast	left to right	3.9 sec
lateral-medium	left to right	7.5 sec
lateral-slow	left to right	14.7 sec
medial-fast	back to front	4.6 sec
medial-medium	back to front	7.9 sec

e.g., the two types of paper (i.e., surfaces 17 and 18) and the two doormats (i.e., surfaces 1 and 5). Some of the surfaces were simply the backsides of other surfaces, such as the red leather (i.e., surface 8) and its backside, the cotton wool (i.e., surface 15). As a control condition, the robot also performed the scratching behaviors when no surface was present (i.e., scratching in mid air), which was recorded as the 21st surface in the dataset. For the rest of the paper, we will use $\mathcal{S}$, where $|\mathcal{S}| = 21$, to denote the set of surfaces that the robot interacted with.

D. Exploratory Behaviors

The robot's set of behaviors, i.e., $\mathcal{B}$, consists of five different exploratory scratching actions. The first three behaviors are *lateral* scratches (i.e., right to left) performed at three different velocities: slow, medium, and fast. The other two behaviors are *medial* scratches (i.e., back to front) performed at two different velocities: fast and medium. In other words, each of the five behaviors is a variation of a prototypical scratching motion, which is attained by varying the direction or the speed at which it is executed. Fig. 4 shows before and after images for two scratching behaviors. The behaviors and their durations are listed in Table I.

The behaviors were encoded as trajectories using the low-level Barrett WAM API (btclient), which uses a proportional-integral-derivative (PID) controller. One lateral trajectory and one medial trajectory were recorded, and their speeds were varied to obtain all five exploratory behaviors. The default velocity parameter for trajectory playback in the Barrett API was used for the medium-velocity setting. The fast velocity

was set by doubling this parameter, while the slow velocity was set by halving it. For all scratching behaviors, the tip of the robot's finger slid across approximately 12 inches of the surface. The behaviors were not designed to maintain constant orientation of the finger relative to the surface as our method does not rely on that. Instead, during the execution of each trajectory an additional 5 N-m torque was applied on joint 1 (shoulder joint) in the downward direction to ensure contact with the surface at all times.

E. Data Collection

Each of the 5 behaviors in $\mathcal{B}$, was performed 10 times by the robot on each of the 21 surfaces in $\mathcal{S}$, which resulted in a total of $5 \times 21 \times 10 = 1050$ behavioral interactions (or trials). During the ith trial, the robot recorded the currently executed behavior $B_i \in \mathcal{B}$, the current surface $S_i \in \mathcal{S}$, and the current accelerometer readings $\mathbf{A_i}$. The sensor readings were represented as $\mathbf{A_i} = [\mathbf{a}^1, \mathbf{a}^2, \ldots, \mathbf{a}^{n_i}]$, where n_i is the number of readings recorded during trial i, and each $\mathbf{a^j} \in \mathbb{R}^3$ denotes the measurements for each of the three accelerometer axes.

To minimize transient noise effects due to wear and tear of the fingernail, the surfaces were swapped throughout the data collection processes. In other words, the surface was changed after the robot scratched it once with all five exploratory behaviors and not scratched again until the robot scratched all other surfaces.[1]

IV. Learning Methodology

A. Feature Extraction

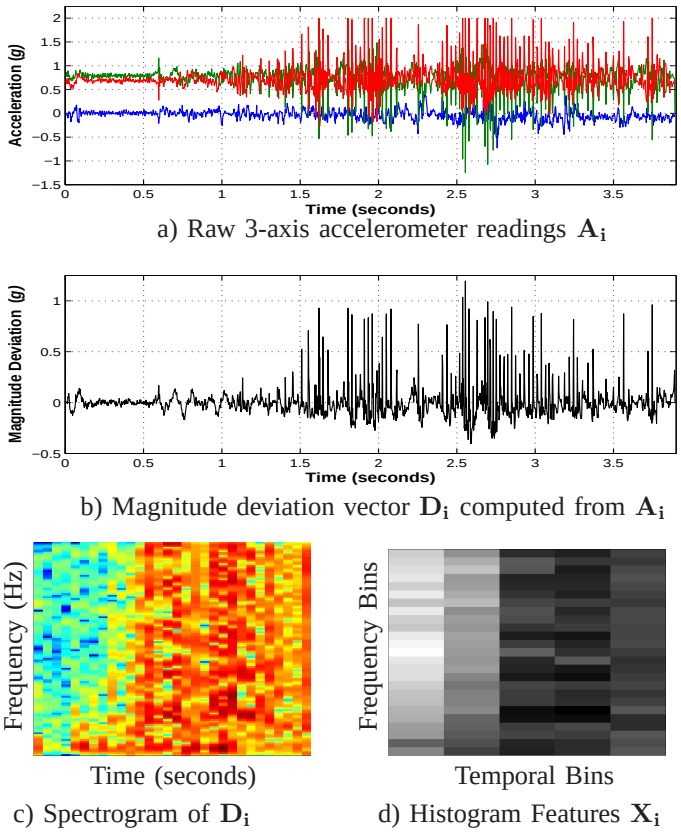
To recognize surfaces based on vibrotactile feedback, some features were first extracted from the accelerometer data. The first step in this process was to convert the accelerometer readings $\mathbf{A_i}$ into a magnitude vector $\mathbf{M_i} = [m^1, m^2, \ldots, m^{n_i}]$, where each $m^j \in \mathbb{R}$ captures the magnitude of the acceleration vector at sample j, while ignoring the direction of the acceleration. This was done using the L2 vector norm given by

$$m^j = |\mathbf{a^j}|_2 = \sqrt{(a_x^j)^2 + (a_y^j)^2 + (a_z^j)^2}$$

The next step was designed to capture the high-frequency components of the acceleration data. To do this, the temporal sequence $\mathbf{M_i}$ was smoothed using a moving average filter with a window of length 100 (i.e., 0.25 s, since the data were recorded at 400 Hz), resulting in the smoothed magnitude acceleration vector $\tilde{\mathbf{M}}_\mathbf{i}$. A magnitude-deviation vector $\mathbf{D_i}$ was computed as $\mathbf{D_i} = \mathbf{M_i} - \tilde{\mathbf{M}}_\mathbf{i}$ in order to capture vibrations detected by the accelerometer.

Fig. 5 shows the raw accelerometer readings $\mathbf{A_i}$ over the course of an interaction, and the resulting magnitude deviation vector $\mathbf{D_i}$. Once $\mathbf{D_i}$ was computed, its discrete Fourier transform with 129 frequency bins was computed, thereby resulting in a spectrogram, which denotes the intensities of different frequencies over time. Fig. 5(c) shows the spectrogram of the magnitude-deviation vector $\mathbf{D_i}$. It should be noted that this is just one way to process the raw accelerometer data (for a review of alternative approaches, see [31]).

[1]A check was performed to verify that sensor wear and tear, if any, was undetectable at the input feature level. If the sensor is changing over time, then two consecutive trials with the same surface and the same behavior should produce feedback signals that are more similar than the ones produced by two trials that are temporally further apart. No such relationship was found.



a) Raw 3-axis accelerometer readings $\mathbf{A_i}$

b) Magnitude deviation vector $\mathbf{D_i}$ computed from $\mathbf{A_i}$

c) Spectrogram of $\mathbf{D_i}$

d) Histogram Features $\mathbf{X_i}$

Fig. 5. Visualization of the feature extraction process used by the robot to recognize surfaces: a) the raw sensor readings $\mathbf{A_i}$ for each accelerometer axis over the course of an interaction with a surface; b) the computed magnitude deviation vector $\mathbf{D_i}$; c) the discrete Fourier transform of $\mathbf{D_i}$ with frequency components in the range of 4-200 Hz; and d) 2-D spectrotemporal histogram features $\mathbf{X_i}$ computed from the spectrogram. In this particular example, the *plastic kitchen roll* (surface 9) was scratched with the *lateral-fast* scratching motion.

While the spectrogram contained a lot of useful information, its dimensionality was fairly high, thereby making it difficult to use with standard machine learning algorithms. To overcome this, a smaller set of spectrotemporal features were extracted using a 2-D spectrotemporal histogram – time (i.e., horizontal axis of the spectrogram) was discretized into k_t bins, while the frequencies (i.e., vertical axis of the spectrogram) were further discretized into k_f bins. In this paper, k_t was set to 5 and k_f was set to 25. Thus, each vibrotactile reading was ultimately represented by a feature vector $\mathbf{X_i} \in \mathbb{R}^{5\times 25}$. Fig. 5 shows the feature-extraction routine, which starts with the accelerometer readings $\mathbf{A_i}$ and ends with the spectrotemporal histogram features $\mathbf{X_i}$.

B. Surface-Recognition Methodology

The primary task of the robot was to recognize the correct surface texture $s \in \mathcal{S}$ given spectrotemporal features $\mathbf{X_i}$ extracted from vibrotactile feedback. To do this, for each behavior $b \in \mathcal{B}$, the robot learned a recognition model $\mathcal{M}_b$ that estimated the surface class, given spectrotemporal features

extracted after performing behavior b on the test surface. More specifically, given features $\mathbf{X_i}$ extracted after performing behavior b on a test surface S_i, the model $\mathcal{M}_b$ estimated the probability $\mathbf{Pr}_b(S_i = s|\mathbf{X_i})$ for all surfaces $s \in \mathcal{S}$.

Two different machine learning algorithms were evaluated as implementations of the models $\mathcal{M}_b$. The first *k*-NN, which is a memory-based algorithm, and which does not build an explicit model of the data [32], [33]. Instead, given a test data point, k-NN finds the k closest neighbors in its training set and outputs a prediction, which is a smoothed average over those neighbors. In this study, the parameter k was set to 3. Class-label probabilities for each surface $s \in \mathcal{S}$ were computed by counting the labels of the k neighbors. For example, if two of the three neighbors had class label *A* then $\mathbf{Pr}(S_i = A) = 2/3$. Similarly, if the class label of the remaining neighbor was *B*, then $Pr(S_i = B) = 1/3$. The k-NN implementation included in the WEKA machine learning library [34] was used to obtain the results.

The second algorithm that we evaluated was an SVM classifier, which is a supervised learning model that falls into the family of *discriminative* models [35]. Given a labeled training set, the SVM algorithm learns a linear decision function that can accurately discriminate between inputs with different class labels. For many problems, however, a good linear separation may not exist in the input space. To resolve this, the labeled inputs can be mapped into a (possibly) higher-dimensional feature space, e.g., $\mathbf{X_i} \rightarrow \Phi(\mathbf{X_i})$, where a good linear decision function can be found. This mapping can be defined implicitly using a kernel function, which specifies how similar two inputs $\mathbf{X_i}$ and $\mathbf{X_j}$ are. In this case, the output of the kernel function $K(\mathbf{X_i}, \mathbf{X_j})$ replaces the dot product $\mathbf{X_i}^\mathbf{T}\mathbf{X_j}$ in the dual-quadratic optimization problem solved by the SVM (for details, see [35] and [36]). In this study, the polynomial kernel function with exponent 2 was used to estimate the similarity between a pair of inputs $\mathbf{X_i}$ and $\mathbf{X_j}$

$$K(\mathbf{X_i}, \mathbf{X_j}) = (\mathbf{X_i}^\mathbf{T}\mathbf{X_j} + 1)^2$$

This kernel function was chosen because it is the most commonly used one in the literature. Other kernel functions (e.g., RBF kernel) available in the WEKA library were also explored, but generally resulted in lower performance. The pairwise-coupling method of Hastie and Tibshirani [37] was applied to generalize the original binary classification SVM algorithm to the multiclass problem of surface recognition. Logistic regression models were fitted to the outputs of the SVM, as described in [34], in order to obtain a probabilistic estimate $\mathbf{Pr}_b(S_i = s|\mathbf{X_i})$ for the surface type of a test data point.

C. Combining Predictions from Different Behaviors

So far, we have described how the robot can learn a surface recognition model $\mathcal{M}_b$ for each behavior $b \in \mathcal{B}$. The robot also needs to be able to efficiently combine predictions from all five models after performing its set of exploratory behaviors on a given surface. To do this, during the training stage, each model $\mathcal{M}_b$ performed its own cross-validation on its training data, and computed a reliability measure w_b, corresponding to the model's estimated accuracy. This procedure allowed the robot to estimate the efficacy of each behavior for solving the surface recognition task.

Next, let $\mathbf{X_1}, \mathbf{X_2}, \ldots, \mathbf{X_N}$ be spectrotemporal features extracted after performing behaviors $b_1, b_2, \ldots, b_N$, respectively, on the test surface $S_{test} \in \mathcal{S}$. Given these data, the robot assigned the prediction to the surface s that maximized:

$$\sum_{i=1}^{N} w_{b_i} \mathbf{Pr}_{b_i}(S_{test} = s|\mathbf{X_i})$$

In other words, given one or more interactions with the same surface, the robot combined the predictions from different behavioral models using estimates for the reliability of each channel of information. The weighted combination of predictions ensures that a model that is not useful for surface recognition will not dominate over other more reliable models.

D. Surface Categorization

In addition to surface recognition, the robot was also tasked with learning surface categories. The method consisted of two steps: 1) estimate a measure of similarity for each pair of surfaces using the surface recognition models and 2) application of unsupervised hierarchical clustering on the surface similarity matrix to construct surface categories.

The intuition behind this approach is that if two or more surfaces are not distinguishable by the robot, then they should be considered similar and placed in the same surface category. To get a measure of similarity for each pair of surfaces, the robot performed *surface-based* cross-validation, i.e., during each iteration, the robot's models were trained on data from $|\mathcal{S}| - 1$ surfaces and tested on the remaining one. Since the test surface was not present in the training set, this procedure forced a recognition error, but it also showed which of the $|\mathcal{S}| - 1$ training surfaces was most similar to the test surface. Let $\mathbf{C}$ be the resulting $|\mathcal{S}| \times |\mathcal{S}|$ confusion matrix such that each entry C_{ij} specifies how often surface i was misclassified as surface j. Thus, each value C_{ij} is an integer in the range of 0-50, since two surfaces can be confused at most 50 times with five behaviors and ten trials per surface. Because most clustering algorithms require the input similarity matrix to be symmetric, another $|\mathcal{S}| \times |\mathcal{S}|$ matrix, i.e., $\mathbf{C}'$, was computed such that each entry $C'_{ij} = 0.5 \times C_{ij} + 0.5 \times C_{ji}$. Finally, the values in $\mathbf{C}'$ were linearly scaled so that each entry was between 0.0 and 1.0, by dividing each value C'_{ij} by 50. As required by the clustering algorithm that was used, the diagonal values of the matrix were set to 1.0. The result of this procedure was a symmetric similarity matrix $\mathbf{W}$, which was used as input to the unsupervised clustering algorithm.

To construct a surface categorization, the robot used the *spectral clustering* algorithm, which falls into the family of *graph-based* or *similarity-based* clustering algorithms [38]. Given a similarity matrix, i.e., $\mathbf{W}$, the algorithm partitions the set of surfaces into disjoint clusters by exploiting the eigenstructure of the matrix $\mathbf{W}$. Because finding an optimal graph partitioning is NP-complete, Shi and Malik [39] proposed an approximation that optimizes the *normalized cut*

TABLE II
SURFACE-RECOGNITION ACCURACY FROM A SINGLE BEHAVIOR

Behavior	k-Nearest Neighbor	Support Vector Machine
lateral-fast	59.5%	64.8%
lateral-medium	52.4%	65.7%
lateral-slow	46.7%	58.6%
medial-fast	43.8%	56.7%
medial-medium	39.5%	45.7%
Average	48.4%	58.3%

objective function. The algorithm, can be summarized with the following steps.

1) Let $\mathbf{W}_{n\times n}$ be the symmetric matrix containing the similarity score for each pair of surfaces.
2) Let $\mathbf{D}_{n\times n}$ be the degree matrix of $\mathbf{W}$, i.e., a diagonal matrix such that $\mathbf{D}_{ii} = \sum_j W_{ij}$.
3) Solve the eigenvalue system $(\mathbf{D} - \mathbf{W})x = \lambda \mathbf{D}x$ for the eigenvector corresponding to the second smallest eigenvalue and use it to bipartition the graph.
4) If necessary, recursively bipartition each subgraph obtained in Step 3.

The algorithm recursively bipartitions the graph induced by the similarity matrix $\mathbf{W}$ until a stopping criterion is reached, thereby producing a hierarchical clustering. The code for the spectral clustering algorithm used in our experiments is listed on the WEKA machine learning repository website (see http://www.cs.waikato.ac.nz/ml/weka/index_related.html). The algorithm is recursively applied until the size of each subgraph falls below five nodes or until the spectral clustering algorithm fails to find a bipartition with a high score according to the normalized cut objective function. The output of this procedure is a hierarchical taxonomy (i.e., a tree), $\mathcal{T}$, which specifies the learned surface categorization.

V. RESULTS

A. Surface Recognition from a Single Behavior

The first experiment evaluated the two machine learning algorithms (k-NN and SVM) on the task of surface recognition from a single behavioral interaction. The results are reported in terms of recognition accuracy

$$\%\ Accuracy = \frac{\#\ correct\ predictions}{\#\ total\ predictions} \times 100$$

The accuracies of the models $\mathcal{M}_b$ were estimated using ten-fold cross validation (i.e., the full dataset was split into ten folds and at each evaluation, nine of those were used for training and one was used for testing).

Table II shows the surface recognition rates for each of the two classification algorithms and for each of the five behaviors. For comparison, given that $|\mathcal{S}| = 21$, a chance classifier is expected to achieve $1/|\mathcal{S}| = 4.76\%$ surface-recognition accuracy. The robot was able to achieve recognition rates substantially better than chance with all five exploratory behaviors. For both lateral and medial scratching behaviors, faster scratching resulted in better model performance. On average across all five behaviors, the SVM algorithm outperformed the k-NN algorithm by roughly 10%. Yet, there is still significant room for improvement.

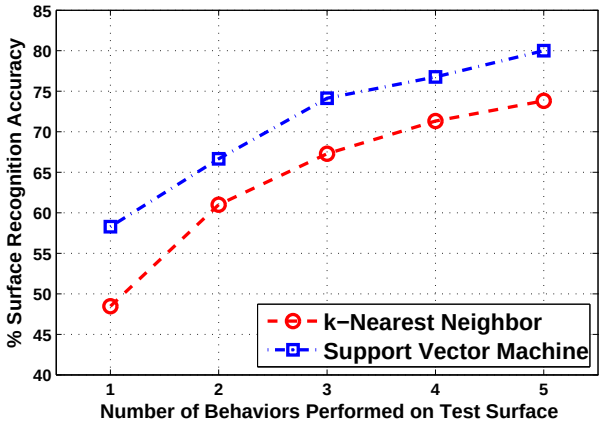


Fig. 6. Surface recognition performance as a function of the number of different exploratory behaviors performed on the test surface. The predictions of the behavior-specific models are combined using a linear weighted combination as described in Section IV-C.

B. Surface Recognition from Multiple Interactions

The next experiment investigated whether the robot can improve its surface recognition rate by performing multiple behaviors on the test surface and then combining the resulting predictions. For example, the robot should be able to recognize a surface with a higher accuracy rate if it performs a sequence of two different behaviors (e.g., fast, followed by medium lateral scratch) than with either behavior alone.

To test this hypothesis, the number of behaviors that the robot performed on a surface at test time was varied from 1 (the default, which is used to generate Table II) to 5 (i.e., performing all five exploratory behaviors on the test surface). When performing two, three and four interactions with the test surface, all possible combinations of behaviors were evaluated and the average recognition rate was recorded. Whenever the robot was performing two or more exploratory behaviors on the test surface, the predictions from the corresponding recognition models were combined, as described in Section IV.C.

Fig. 6 shows the recognition rates for this experiment with the k-NN and SVM learning algorithms. As more behaviors are performed on the test surface, the recognition rate increases dramatically. Using SVM, the recognition rate increases to 80.0% after performing all five exploratory behaviors. This apparent boost of the recognition rate is consistent with previous results on interactive object recognition tasks, which also show that applying multiple behaviors can greatly improve object recognition accuracy [40], [41], [42]. Therefore, robots can achieve higher tactile recognition rates if they can perform different types of scratching behaviors, as opposed to just one.

An additional experiment was performed to test if the same type of recognition boost can be obtained by combining two instances of the *same* scratching behavior instead of two *different* scratching behaviors, which are all performed on the same surface. To do this, the dataset was split into five folds, each containing two trials with all five behaviors per surface. During each round of evaluation, the model was trained on

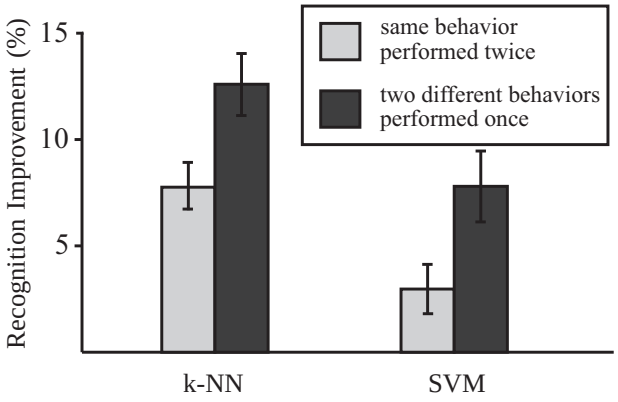


Fig. 7. Comparison of the improvement in classification accuracy when combining feedback from two executions of the *same* behavior versus two executions of *different* behaviors. Regardless of the learning algorithm (i.e., k-NN or SVM), the classification improvement is higher when combining feedback from two distinct scratching behaviors.

four of these folds (i.e., eight trials) and tested on the test fold (i.e., two trials). Next, the model's outputs were computed for all five possible combinations of the same behavior, as well as for all $\binom{10}{2} - 5 = 40$ possible combinations of different behaviors present in the test set. These outputs were compared with the true labels and used to estimate the improvement in recognition accuracy, as described below.

Let $acc(B_i, B_j)$ be the expected recognition accuracy when combining feedback from behaviors B_i and B_j, and let $acc(B_i)$ and $acc(B_j)$ be their individual accuracies. The recognition improvement (i.e., RI_{ij}) obtained when using any two behaviors B_i and B_j (which could be the same or different) can be measured relative to the classification performance of the individual behaviors, i.e.,

$$RI_{ij} = acc(B_i, B_j) - \frac{acc(B_i) + acc(B_j)}{2}$$

This formulation allows us to test if combining feedback from two different behaviors results in greater recognition boost than combining feedback from two executions of the same behavior. The results of this evaluation are shown in Fig. 7. The average recognition improvement when using the same behavior was estimated from five samples, i.e., one for each of the five scratching behaviors. The improvement attained when combining two different behaviors was estimated from ten samples, i.e., one for each unique pair of behaviors. For both learning algorithms, the classification improvement is higher when two different behaviors are applied on the test surface, as opposed to applying the same behavior twice.

C. Recognition Errors

It is also worth investigating the type of recognition errors that the robot makes. Fig. 8 shows the confusion matrix obtained when using the SVM learning algorithm and applying all five behaviors on the test surface. The confusion matrix indicates how often a given surface was misclassified as another surface (perfect classification would result in a diagonal matrix). In this case, the overall accuracy is 80.0%. The matrix

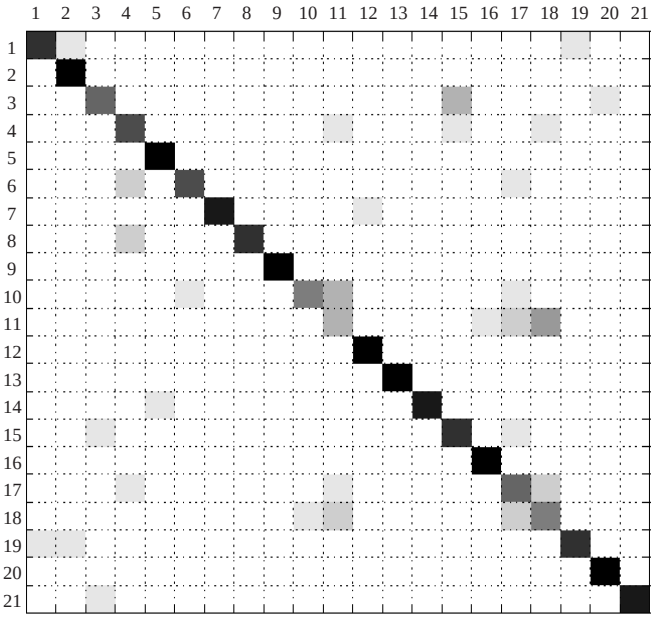


Fig. 8. Confusion matrix obtained after evaluating the robot's recognition model when all five exploratory behaviors were applied on the test surface. The matrix specifies how often a surface was classified correctly (i.e., diagonal entries) or mis-classified as another surface (i.e., off diagonal entries). Dark colors indicate high values, while light colors indicate low values.

shows that most errors involve pairs of surfaces with similar textures.

For example, the *white paper* (i.e., surface 17) and the *yellow paper* (i.e., surface 18) are often confused with each other. This is to be expected, however, as the two papers are almost identical. Similarly, the *soft cloth* (i.e., surface 3) and the *cotton wool cloth* (i.e., surface 15) are also confused with each other. Surfaces that are unique within the dataset (e.g., surface 9, the *plastic kitchen roll*) are generally recognized with higher accuracy. The results also show that the *table* surface (i.e., surface 10) is often confused with the two types of paper, as well as the *bed sheet* (i.e., surface 11). This is likely due to the fact that those surfaces are rather thin, and thus, some of the detected tactile feedback is due to the table surface on which they were placed.

Overall, the confusion matrix shows that the errors of the robot's recognition models are not random in nature. Instead, whenever an error is made, the predicted surface is often somewhat similar to the actual one, in terms of material and/or texture. This suggests that the recognition models could be used to estimate a measure of similarity between surfaces based on vibrotactile data.

D. Surface Categorization

In the next experiment, the robot used its surface-recognition models to learn surface categories. Let $\mathbf{W}$ be the resulting object similarity matrix after performing cross-validation as described in Section IV-D. The similarity matrix can be visualized in two dimensions by converting it into a distance matrix and embedding it onto the 2-D plane using the ISOMAP method for non-linear dimensionality reduction [43].

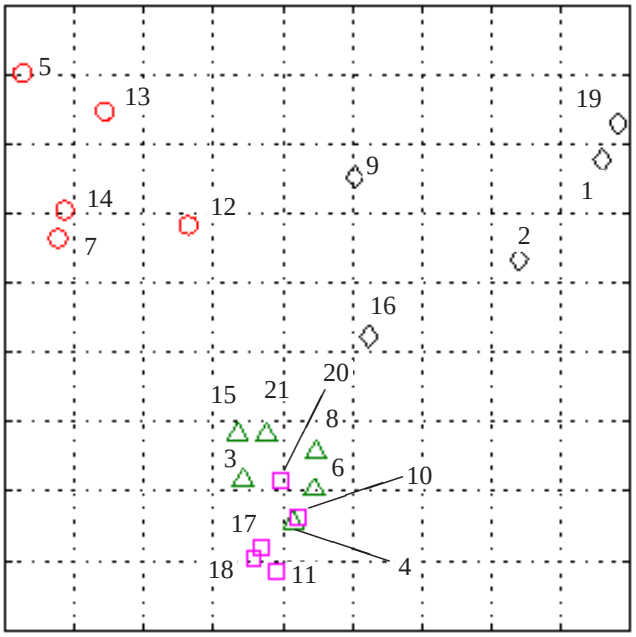


Fig. 9. Two-dimensional embedding of the pair-wise surface-similarity matrix used for surface categorization. The two axes correspond to the first two dimensions in the lower dimensional embedding computed by the ISOMAP algorithm, and thus, do not have physical units. Surfaces with high similarity in $\mathbf{W}$ appear close to each other in the graph. The glyph (color and shape) for each surface corresponds to the category of the surface according to a particular cut through the learned surface hierarchy (*cut 2* in Fig. 10).

The 2-D embedding of the similarity matrix is shown in Fig. 9. The closeness between surfaces in the ISOMAP embedding corresponds to their similarity in $\mathbf{W}$, i.e., surfaces with high similarity appear close to each other in the graph. Like the confusion matrix (see Fig. 8), the embedding shows which surfaces are considered to be similar from the point of view of the robot's recognition models.

To construct surface categories, spectral clustering was applied on the similarity matrix $\mathbf{W}$, as described in Section IV-D. The resulting hierarchical surface categorization is shown in Fig. 10. Three cuts through the hierarchy are also shown: *cut 1*, which corresponds to the initial top-level split; *cut 2*, which was produced by further splitting the two top-level clusters; and *cut 3*, which corresponds to the leaf cut of the hierarchy. It is important to note that the robot's experience with the set of surfaces was still quite limited, and thus, it is unrealistic to expect that the learned surface categorization would perfectly match a categorization provided by a human.

Upon closer examination, however, most leaf clusters in the learned hierarchy tend to consist of highly similar surfaces. For example, the first leaf category contains the *soft cloth* and the *soft wool cloth*, as well as the 21st surface in the dataset (scratching in mid air). Both leathers (i.e., surfaces 4 and 8) are placed in the second cluster, along with the *bulletin board* (i.e., surface 6). The third leaf cluster contains surfaces that are highly similar to the bare table (i.e., surface 10). For example, the two papers (i.e., 17 and 18) contained in that same cluster are so thin that the detected tactile feedback is largely influenced by the table on which they are laid. Both

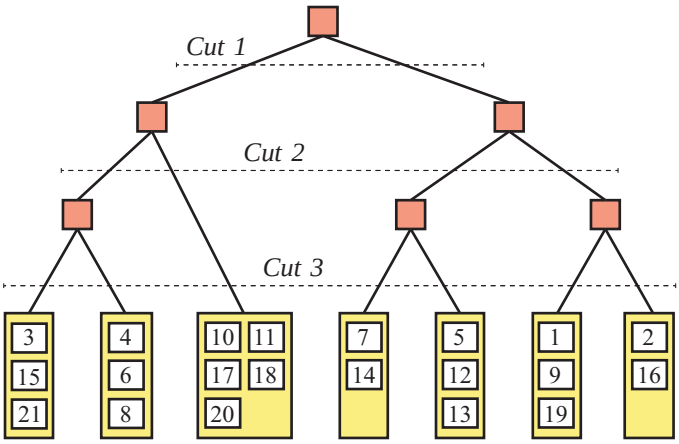


Fig. 10. The learned hierarchical surface categorization. Three possible cuts through the hierarchy are shown: *Cut 1*, corresponds to the top-level split into two categories; *cut 2* is produced by further splitting the two top-level clusters; *cut 3* corresponds to the leaf cut of the hierarchy.

the front side of the thinner doormat (i.e., surface 5) and its backside (i.e., surface 13) end up in the same cluster. On the other hand, the front and back sides of the corduroy (i.e., surfaces 7 and 12) end up in two different leaf clusters, which share the same parent cluster.

E. Surface-Category Recognition

The robot was also evaluated on how well it can recognize the correct category of a *novel* surface (i.e., a surface that was not present in the training set). Given a hierarchical surface categorization and a chosen cut through the hierarchy, SVM models were trained that could accurately label data from a novel surface with the correct surface category. For example, given the hierarchy shown in Fig. 10 and cut 1, the robot's task was to label a novel surface as belonging to either one of the top two high-level categories. Once the prediction was made, it was compared with the actual category of the novel surface (i.e., the category to which the surface would have been assigned if the robot had performed all five behaviors ten times on this surface before estimating the surface categorization).

Category-recognition rates were estimated using *surface-based* cross-validation: At each iteration, data from $|\mathcal{S}| - 1$ surfaces were used for training the model, while data from the remaining one surface was used for testing. The hierarchical categorization itself remained fixed in order to compare the predicted surface category to the actual one (i.e., the one shown in Fig. 10). In other words, the elements in the surface similarity matrix $\mathbf{W}$ corresponding to the novel surface were only used when evaluating the outputs of the recognition model and not when training it.

The results of this experiment are shown in Fig. 11. Overall, the robot was able to recognize the category of a novel surface with accuracy substantially better than chance. Performance was best with more abstract or high-level surface categories (e.g., *cut 1*) and worst with the lowest level cut (i.e., the leaf cut). As expected, the accuracy rates for *cut 3* are worse than the surface recognition rates (see Fig. 6), since in the category-recognition experiment, the robot is evaluated on

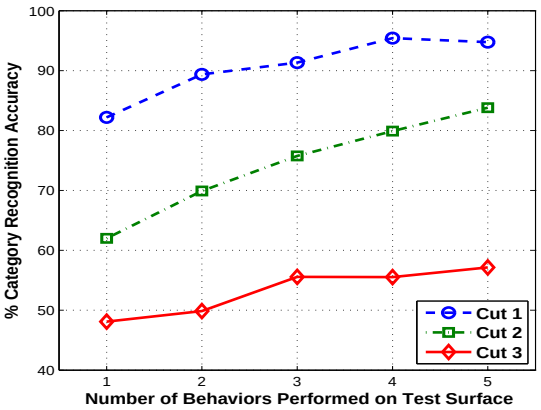


Fig. 11. Surface-category-recognition rates using SVM for three different cuts through the learned hierarchy. In this experiment, the robot was tasked with recognizing the correct category of a novel surface (i.e., one that was not available in the training set).

novel surfaces, as opposed to familiar ones. Nevertheless, the recognition performance for all three cuts improved as more exploratory behaviors were performed on the test surface.

To summarize, the results from the surface categorization experiment in Section V-D showed that the robot was able to learn surface-recognition models and use them to construct a hierarchical surface categorization. The results from this section show that the robot was also able to learn SVM models that can label a novel surface with its correct surface category. Overall, the learned hierarchical clustering of the surfaces captures some of their physical properties and, generally, tends to group similar surfaces into the same cluster of the hierarchy.

VI. Conclusion

This paper presented a method and a representation for surface recognition and surface categorization using a three-axis fingertip accelerometer. The sensor was mounted on one of the fingers of our humanoid robot and was able to capture the vibrations characteristic of a given surface as the robot scratched it. Our robot was programmed with five different exploratory scratching behaviors, which it used to explore 20 different surfaces that were made of various materials (e.g., cloth, wood, paper, leather, etc.). Using frequency-domain analysis and spectrotemporal features, the robot was able to recognize the surfaces with 80.0% accuracy. Applying multiple different types of scratching behaviors on a test surface resulted in higher recognition rates than with any single behavior alone.

The robot's recognition model was also used to estimate a hierarchical surface categorization, which groups similar surfaces together. Learning abstract, but meaningful, categories of surfaces (or more generally, objects) is a prerequisite for an intelligent robot expected to handle the hundreds of objects present in our homes and offices. A hierarchical representation of surface types can potentially allow a robot to characterize and recognize an even larger number of surfaces. The hierarchical representation can also be used by the robot to characterize a novel surface (i.e., one that is not available in the training set) by associating it with the most-similar cluster of surfaces in the learned hierarchy.

One of the major findings of this study is that the use of multiple exploratory behaviors can be crucial for improving the robot's recognition rate. Performing the same behavior twice also improved the accuracy, but the improvement was larger when two different behaviors were used. Given the large potential space of scratching trajectories, a potentially fruitful direction for future work may involve automated learning of useful behaviors and exploratory procedures by a robot for the task of characterizing and recognizing surfaces. Such an ability is important, especially considering that the optimal set of behaviors depends on the surfaces explored by the robot and, thus, cannot be preprogrammed in advance. In addition to learning optimal behaviors, it would also be desirable to implement bi-manual scratching behaviors (e.g., hold an object with one hand and scratch it with the other) in order to scale up the existing method to a wider variety of household objects.

Integrating other modalities (e.g., audio, vision, and proprioception) into the framework is also a viable direction for future work. Audio, in particular, would allow the robot to perceive surface properties by performing exploratory behaviors such as tapping on the surface and listening for feedback. Some of our research results already indicate that integrating vibrotactile feedback with proprioception (in the form of joint-torque values over the course of the interaction) results in even higher surface-recognition rates than the ones obtained when using either sensory modality alone [42].

While the number of surfaces used in this study was larger than those used in previous related work on tactile recognition, it still pales in comparison to the hundreds of different types of surfaces present in our homes and offices. A hierarchical surface categorization is one possible way to handle such a large number of entities. In future work, the robot can be evaluated on learning compact surface and object hierarchies with emphasis on integration of novel entities into the learned categorization. Robots that can learn about large sets of objects and their relationships will undoubtedly be better suited to handle the multiple challenges of human-inhabited environments.

References

[1] R. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile Sensing From Humans to Humanoids," *IEEE Transactions on Robotics*, vol. 26, pp. 1–20, 2010.

[2] M. Hollins and S. Bensmaïa, "The coding of roughness," *Canadian Journal of Experimental Psychology*, vol. 61, no. 3, pp. 184–195, 2007.

[3] R. Johansson and J. Flanagan, "Coding and use of tactile signals from the fingertips in object manipulation tasks," *Nature Reviews Neuroscience*, vol. 10, pp. 345–359, May 2009.

[4] S. Lederman and R. Klatzky, "Haptic Classification of Common Objects: Knowledge-driven exploration," *Cognitive Psychology*, vol. 22, pp. 421–459, 1990.

[5] T. G. Power, *Play and Exploration in Children and Animals*. Mahwah, NJ: Lawrence Erlbaum, 2000.

[6] H. A. Ruff, "Infants' manipulative exploration of objects: Effects of age and object characteristics." *Developmental Psychology*, vol. 20, no. 1, pp. 9–20, 1984.

[7] D. Stack and M. Tsonis, "Infants' haptic perception of texture in the presence and absence of visual cues," *British Journal of Developmental Psychology*, vol. 17, pp. 97–110, 1999.

[8] W. M. Bergmann-Tiest and A. M. L. Kappers, "Haptic and visual perception of roughness," *Acta Psychologica*, vol. 124, no. 2, pp. 177 – 189, 2007.

[9] D. Lynott and L. Connell, "Modality exclusivity norms for 423 object properties," *Behavior Research Methods*, vol. 41, no. 2, 2009.
[10] S. Lederman and R. Klatzky, "Hand movements: a window into haptic object recognition," *Cognitive Psychology*, vol. 19, pp. 342–368, 1987.
[11] G. Cannata, M. Maggiali, G. Metta, and G. Sandini, "An embedded artificial skin for humanoid robots," in *Proc. of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2008.
[12] G. Cannata, R. S. Dahiya, M. Maggiali, F. Mastrogiovani, G. Metta, and M. Valle, "Modular skin for humanoid robot systems," in *Proc. of the 4th International Conf. on Cognitive Systems, Zurich. Switzerland*, 2010.
[13] B. Argall, E. Sauser, and A. Billard, "Tactile guidance for policy refinement and reuse," in *Proceedings of the IEEE International Conference on Development and Learning (ICDL), Ann Arbor, MI*, Aug 18-21 2010.
[14] ——, "Policy adaptation through tactile correction," in *Proceedings of the 36-th Annual Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB'10)*, 2010.
[15] M. Tanaka, J. Levequem, H. Tagami, K. Kikuchi, and S. Chonan, "The "haptic finger" a new device for monitoring skin condition," *Skin Research and Technology*, vol. 9, no. 1, pp. 131–136, 2003.
[16] R. Howe and M. Cutkosky, "Dynamic tactile sensing: perception of fine surface features with stress rate sensing," *Robotics and Automation, IEEE Transactions on*, vol. 9, no. 2, pp. 140–151, 1993.
[17] K. Hosoda, Y. Tada, and M. Asada, "Anthropomorphic robotic soft fingertip with randomly distributed receptors," *Robotics and Autonomous Systems*, vol. 54, no. 2, pp. 104 – 109, 2006.
[18] Y. Mukaibo, H. Shirado, M. Konyo, and T. Maeno, "Development of a texture sensor emulating the tissue structure and perceptual mechanism of human fingers," in *Proc. of the 2005 IEEE International Conference on Robotics and Automation (ICRA)*, April 2005, pp. 2565–2570.
[19] G. Cannata and M. Maggiali, "Processing of tactile/force measurements for a fully embedded sensor," in *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2006.
[20] N. Wettels, V. Santos, R. Johansson, and G. E. Loeb, "Biomimetic tactile sensor array," *Advanced Robotics*, vol. 22, pp. 829–849, June 2008.
[21] R. S. Dahiya, G. Metta, and M. Valle, "Development of fingertip tactile sensing chips for humanoid robots," in *Proc. of the 2009 IEEE International Conference on Mechatronics*, 2009.
[22] N. Jamali and C. Sammut, "Material classification by tactile sensing using surface textures," in *Proc. of the 2010 IEEE International Conference on Robotics and Automation (ICRA)*, May 2010, pp. 2336 –2341.
[23] E. H. Adelson, "Image statistics and surface perception," *Human Vision and Electronic Imaging XIII, Proceedings of the SPIE*, vol. 6806, pp. 680 602–680 602–9, 2008.
[24] Y. Tanaka, M. Tanaka, and S. Chonan, "Development of a sensor system for collecting tactile information," *Microsyst. Technol.*, vol. 13, pp. 1005–1013, April 2007.
[25] L. Beccai, S. Roccella, A. Arena, F. Valvo, P. Valdastri, A. Menciassi, M. C. Carrozza, and P. Dario, "Design and fabrication of a hybrid silicon three-axial force sensor for biomechanical applications," *Sensors and Actuators A: Physical*, vol. 120, no. 2, pp. 370 – 382, 2005.
[26] F. de Boissieu, C. Godin, C. Serviere, and D. Baudois, "Tactile texture recognition with a 3-axial force mems integrated artificial finger," in *Proceedings of the 2009 Robotics Science and Systems Conference (RSS), Seattle, WA*.
[27] J. Romano, T. Yoshioka, and K. Kuchenbecker, "Automatic filter design for synthesis of haptic textures from recorded acceleration data," in *Proceedings of the IEEE International Conference on Robotics and Automation, Anchorage, AK*, 2010, pp. 1815–1821.
[28] K. Kuchenbecker, "Haptography: capturing the feel of real objects to enable authentic haptic rendering," in *Proceedings of Haptic in Ambient Systems (HAS) Workshop, Quebec City, Canada*, Feb 2008.
[29] R. Howe and M. Cutkosky, "Sensing skin acceleration for slip and texture perception," in *Proceedings of the IEEE International Conf. on Robotics and Automation*, 1989, pp. 145–150.
[30] V. Sukhoy, R. Sahai, J. Sinapov, and A. Stoytchev, "Vibrotactile recognition of surface textures by a humanoid robot," in *Proceedings of the Humanoids 2009 Workshop "Tactile Sensing in Humanoids - Tactile Sensors and Beyond", Paris, France*, Dec 7, 2009, pp. 57–60.
[31] N. Landin, J. Romano, W. McMahan, and K. Kuchenbecker, "Dimensional reduction of high-frequency accelerations for haptic rendering," in *Haptics: Generating and Perceiving Tangible Sensations*, ser. Lecture Notes in Computer Science, A. Kappers, J. van Erp, W. B. Tiest, and F. van der Helm, Eds. Springer Berlin / Heidelberg, 2010, vol. 6192, pp. 79–86.
[32] W. D. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithm," *Machine Learning*, vol. 6, pp. 37–66, 1991.
[33] C. G. Atkeson, A. W. Moore, and S. Schaal, "Locally weighted learning," *Artificial Intelligence Review*, vol. 11, no. 1-5, pp. 11–73, 1997.
[34] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*, 2nd ed. San Francisco: Morgan Kaufman, 2005.
[35] V. Vapnik, *Statistical Learning Theory*. New York: Springer-Verlag, 1998.
[36] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.
[37] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," *Advances in Neural Information Processing Systems*, vol. 10, 1998.
[38] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.
[39] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.
[40] J. Sinapov, M. Weimer, and A. Stoytchev, "Interactive learning of the acoustic properties of household objects," in *Proceedings of the IEEE International Conference on Robotics and Automation, Kobe, Japan*, 2009, pp. 2518–2524.
[41] T. Bergquist, C. Schenck, U. Ohiri, J. Sinapov, S. Griffith, and A. Stoytchev, "Interactive object recognition using proprioceptive feedback," in *Proc. of the IROS Workshop "Semantic Perception for Mobile Manipulation", St. Lois, MO, Oct. 15, 2009*.
[42] J. Sinapov and A. Stoytchev, "The boosting effect of exploratory behaviors," in *Proc. National Conferrence on Artificial Intelligence (AAAI), Atlanta, GA*, July 11-15 2010, pp. 1613–1618.
[43] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

Jivko Sinapov (S'09) received the B.S. degree in Computer Science from the University of Rochester, NY, in 2005. He is currently working towards a PhD degree in Computer Science with the Developmental Robotics Laboratory, Iowa State University, Ames. His current research interests include developmental robotics, robotic perception, manipulation, and machine learning.

Vladimir Sukhoy received the Bachelor's degree in applied mathematics from Donetsk National University, Donetsk, Ukraine, in 2004. He is currently working toward the PhD degree in Computer Engineering with the Developmental Robotics Laboratory, Iowa State University, Ames. His current research interests are in the areas of developmental robotics, human-computer interaction, computational perception, and machine learning.

Ritika Sahai received the B.S. degree in Electrical Engineering from Iowa State University, Ames in 2010. She is currently a Master's student in Computer Engineering at Iowa State University. Her research interests are in the areas of developmental robotics, learning the affordances of writing tools, and robotic perception of surface textures.

Alexander Stoytchev (S'00-M'07) received the M.S. and Ph.D. degrees in computer science from the Georgia Institute of Technology, Atlanta in 2001 and 2007, respectively. He is currently an Assistant Professor of Electrical and Computer Engineering and the Director of the Developmental Robotics Laboratory, Iowa State University, Ames. His current research interests are in the areas of developmental robotics, autonomous robotics, computational perception, and machine learning.