Modeling and Rendering Realistic Textures from Unconstrained Tool-Surface Interactions

Heather Culbertson, *Student Member, IEEE*, Juliette Unwin, and Katherine J. Kuchenbecker, *Member, IEEE*

Abstract—Texture gives real objects an important perceptual dimension that is largely missing from virtual haptic interactions due to limitations of standard modeling and rendering approaches. This paper presents a set of methods for creating a haptic texture model from tool-surface interaction data recorded by a human in a natural and unconstrained manner. The recorded high-frequency tool acceleration signal, which varies as a function of normal force and scanning speed, is segmented and modeled as a piecewise autoregressive (AR) model. Each AR model is labeled with the source segment's median force and speed values and stored in a Delaunay triangulation to create a model set for a given texture. We use these texture model sets to render synthetic vibration signals in real time as a user interacts with our TexturePad system, which includes a Wacom tablet and a stylus augmented with a Haptuator. We ran a human-subject study with two sets of ten participants to evaluate the realism of our virtual textures and the strengths and weaknesses of this approach. The results indicated that our virtual textures accurately capture and recreate the roughness of real textures, but other modeling and rendering approaches are required to completely match surface hardness and slipperiness.

Index Terms—Haptic texture rendering, virtual reality, tablet computers, data-driven modeling, high-frequency vibrations

1 Introduction

When you use a tool to touch real objects, the vibrations your fingertips experience convey important information about the interaction. These vibrations are caused by contact between the tooltip and small surface features [1] and are transmitted to your hand through the rigid link of the tool. For example, if you pick up a pen and drag it across a table, a rock, or a swatch of fabric, you can feel variations in the texture even though your finger is not directly touching the surface. The human sense of touch excels at sensing and interpreting these vibrations to gather information about the physical world [2]. Unfortunately, the richness of these interaction cues is missing from many virtual environments, leading to a less satisfying and immersive experience.

Feeling appropriate tool vibrations during virtual interactions can create the perceptual illusion that one is touching a real surface. However, most modern haptic algorithms and devices are incapable of outputting high-fidelity reproductions of such vibrations [3], [4]. Many approaches to representing textures have been proposed, but no consensus exists [5]. However, measurement-based models can be significantly more realistic than models that are simply hand-tuned without a basis in physical reality [6]. Recent research has proven the technological feasibility of creating texture models from recorded high-frequency tool acceleration signals [7], [8], [9].

This paper presents and evaluates a new method for the creation of haptic texture models from short acceleration, force, and speed signals recorded when an individual explores a textured surface with an instrumented tool in a natural and unconstrained manner. First presented in [10], this method is efficient in both required data and computation time. We focus particularly on the vibrations caused by tool-surface contact; these signals relate to but do not completely capture the hardness and slipperiness of the surface. Thus, our data-driven approach represents a carefully chosen practical simplification of the tool-surface interaction.

This paper aims to evaluate our approach to texture modeling and rendering. Although much research has been done in determining the perceptual qualities of real textures [11], [12], [13], there has not been extensive work in evaluating how these perceptual qualities translate to virtual textures. Past studies that have analyzed virtual textures focused on realism as a whole [8], [14]. In contrast, this work seeks to assess the perceptual qualities and similarities of real and virtual textures to evaluate the strengths and weaknesses of our modeling approach. We hope this investigation helps researchers make informed decisions about adopting similar approaches.

2 Background

Modeling and rendering realistic tool-mediated textures is a complex and multi-faceted problem that has not been fully solved. One of the limitations to finding a solution to this problem is the inherent complexity of interactions between a tool and the surface. Complete physics-based simulations are too computationally complex for real-time rendering [5]. A physics-based simulation that was recently created to explain the transients caused by a tool contacting a textured

- *H. Culbertson and K. J. Kuchenbecker are with the Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia, PA 19104. E-mail: {hculb, kuchenbe}@seas.upenn.edu.*
- *J. Unwin is with the Department of Mechanical Engineering, University of Southampton, Southampton SO17 1BJ, United Kingdom. E-mail: ettie.unwin@gmail.com.*

Manuscript received 31 Jul. 2013; revised 19 Jan. 2014; accepted 1 Apr. 2014; date of publication xx xx xxxx; date of current version xx xx xxxx.
Recommended for acceptance by W. Michiel.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TOH.2014.2316797

surface showed a promising match to data recorded from a physical surface [15]. However, it could not be completed in real time and would thus not be applicable to rendering virtual textures.

To lower the complexity to a point where real-time execution is feasible, haptic texture rendering schemes seek to model textures via a simplification of the underlying physics. Many attempts have been made to recreate these interactions, but most have not been able to completely match the richness and usefulness of haptic feedback experienced during interactions with the physical world. One early approach to modeling textures involved the implementation of virtual springs to perturb the height of the user's hand [16]. This rendering approach was also applied to generate stochastic textures using Gaussian-distributed force fields [17]. However, the goal of these two rendering approaches was to produce distinguishable textures and not textures with a real-world basis.

An early approach that did have a basis in real textures analyzed the pixels of a grayscale image to create a surface height profile that directly depended on shading value [18]. Data-driven modeling schemes such as this have an advantage over physics-based modeling in that they do not require tedious and potentially inaccurate hand-tuning. Rather, data-driven models seek to capture the output response of a system (e.g., force, acceleration) given user inputs (e.g., position, velocity, force) [6].

Several researchers have explored the use of data captured from real interactions to increase the realism of virtual haptic interactions. Past applications include the use of real-world data to virtually represent the sensations felt when cutting [19] and tapping [4]. Data-driven modeling has also recently been used in haptic augmented reality, which seeks to modulate the haptic properties of real objects using virtual feedback [20], [21].

Several approaches have applied the principles of data-driven modeling in the area of haptic textures. A variety of data types have been explored, with the most common being vibrations induced when dragging a tool across a textured surface [7], [8], [9], [10], [22], [23]. Direct playback of recorded vibrations has also been implemented using voice-coil actuators [24] and a cable-driven system [14]. Other types of data used to model textures include friction variations [25] and vertical perturbations [26] resulting from dragging across a surface.

From the many types of data recorded from interactions with physical surfaces, the use of tool vibrations has resulted in the most realistic haptic textures to date [8]. We expanded this modeling approach to use data that was recorded in a natural manner without constraints on a user's force, speed, or motion [10]. This article further refines our approach and evaluates its ability to create artificial haptic textures on different perceptual dimensions via an extensive study.

3 Data Collection

The first step in creating a haptic texture model is to record data from tool interactions with real textures. This section details the hardware and calculations needed to record and process the interaction data. Then we present and discuss the specific surfaces modeled in this paper.

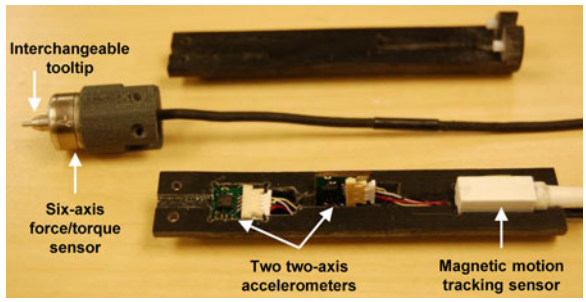

Fig. 1. The haptic recording device measures position, orientation, force, and vibration for three axes.

3.1 Hardware

We capture tool-surface interaction data using the custom haptic recording device first presented in [10] and shown in Fig. 1. This device records the tool's position, orientation, contact force, and high-frequency acceleration while it is dragged across a textured surface.

The 8 mm Ascension 3D Guidance TrakSTAR magnetic tracking sensor embedded in the back of the tool measures the tool's position and orientation at 125 Hz with a resolution of 0.5 mm and 0.1 degrees respectively. The force sensor is a Nano17 SI-25-0.25 six-axis force/torque transducer from ATI Industrial Automation Inc. A stainless steel hemispherical tip with a diameter of 1.6 mm was attached to the force sensor. A 16-bit USB analog-to-digital DAQ device from National Instruments records the force and acceleration data at a rate of 10 kHz.

Numerous three-axis analog accelerometers were considered to measure the tool vibrations, but none fit the required size, range, and bandwidth specifications. Consequently two high-bandwidth two-axis ± 18 g ADXL321 analog accelerometers were chosen, having a resonant frequency of 5.5 kHz and resolution of 0.126 m/s^2. These accelerometers are rigidly embedded in the tool at right angles to allow for measurement of three orthogonal axes plus a fourth redundant axis. The accelerometers are band-limited by a five-pole low-pass Bessel filter at 1,000 Hz to remove the effects of sensor resonance and frequencies above the bandwidth of human vibration detection [27]. Although humans cannot feel these higher frequencies, they can affect the feel of the surface because the channels that process auditory and touch frequency contents are linked [28]. Phase distortion, which is an artifact of the modeling approach taken in this work, does not directly affect the feel of textures because humans cannot discriminate phase shifts in haptic signals [29]. However, it could still interfere with the tactile perception of the surface for this study because it alters the way that synthesized vibrations sound.

3.2 Data Recording

Data is recorded while the user holds the haptic recording device and explores a selected surface using natural and unconstrained motions. The user is free to vary normal force and scanning speed during each ten-second recording; however, a range of normal forces and scanning speeds should be used to best capture the feel of the surface under different contact conditions. A single ten-second recording was made for each material presented.

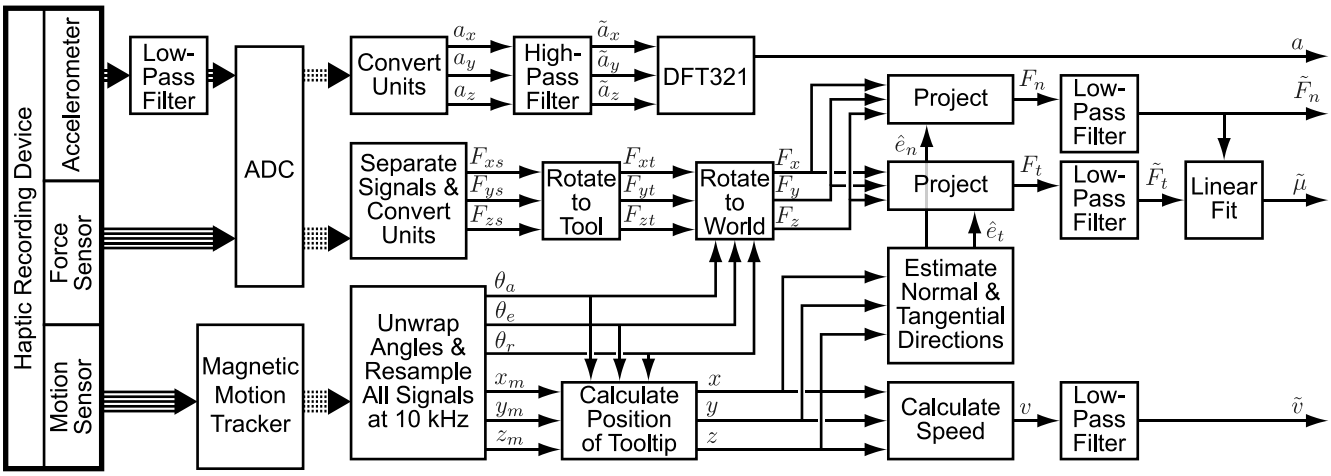

Fig. 2. A diagram of our data recording and processing system. Acceleration and force are recorded at 10 kHz. Position and orientation are recorded at 125 Hz and upsampled to 10 kHz. A rotation matrix calculated from the orientation data at each time step is used to find the force vector and tooltip position vector in the world frame. The normal and tangential forces and tooltip speed are calculated and low-pass filtered.

3.3 Data Processing

The steps required to transform the recorded data into data usable for modeling are shown in Fig. 2. The three acceleration signals are first converted from volts to m/s^2. A high-pass filter at 20 Hz is applied to the three signals to remove gravity and the effects of purposeful human motion. Since humans cannot discriminate the direction of high-frequency vibrations [27], the three filtered signals are mapped onto a single axis using the DFT321 algorithm, which preserves the spectral and temporal properties of the three-axis signals [30].

The force signals are separated into the three force axes, and a constant rotation matrix is applied to rotate these force signals to the tool frame. The Euler angles and the positions are upsampled to 10 kHz to be at the same rate as the force measurements. A rotation matrix from the tool frame to the world frame of the magnetic tracker's field generator is calculated at each time step. The forces are rotated to the world frame, and the tooltip position in the world frame is found by adding the sensor position in the world frame to a rotated constant offset vector from the tracker sensor to the tooltip (obtained through calibration).

After recording, a plane is fit to the tooltip positions in the world frame to approximate the surface of the material. At each time step, the unit vector normal to the surface plane ($\hat{e}_n$) and the unit vector in the direction of motion ($\hat{e}_t$) are calculated. The inner product is calculated between $\hat{e}_n$ and the force vector in the world frame $\vec{F}$ to find the normal force (F_n), and separately the inner product is calculated between $\hat{e}_t$ and $\vec{F}$ to find the tangential force (F_t). F_n and F_t are used to calculate the friction coefficient for the surface, as described in Section 3.4.3. The tooltip speed v is calculated as the magnitude of the discrete-time derivative of the tooltip position vector. The normal force, tangential force, and speed are low-pass filtered at 100 Hz to reduce noise.

3.4 Texture Samples

The six surfaces modeled in this paper (rough plastic, canvas, floor tile, silk, vinyl, and wood) are the same as those in [10]. Shown in Fig. 3, they were chosen to create a set that is diverse in roughness, hardness, slipperiness, and fineness. All materials were cut to 10.2 cm squares. The cloth materials (canvas, silk, vinyl) were mounted on a 0.32 cm thick piece of acrylic using double-sided tape. Although a limited set of materials was chosen for presentation in this work, the modeling and rendering methods are valid for a wide range of isotropic and homogeneous materials, as demonstrated in [31], [32].

Although they were not used directly in modeling, we determined the physical parameters of the materials so they could be compared to the rated parameters in the study reported later in this paper. The parameters of interest were roughness (surface feature height), hardness (stiffness), slipperiness (friction), and fineness (spatial frequency of surface features).

3.4.1 Roughness Measurements

The roughness of each material was found optically using the images shown in Fig. 3. The analyzed images were square with 500 pixels on each edge, which correlated to a physical size of 63.5 mm. All pictures were taken from overhead using a Nikon D40 digital camera mounted on a tripod under fluorescent lighting. The images were converted to grayscale and separated into their roughness and waviness components using a 2D Gaussian low-pass filter ($\sigma =$ 2.54 mm). In the roughness component, a light pixel

Fig. 3. Six textures that were haptically modeled with the presented methods. T6* was modeled but excluded from the study due to its anisotropy.

TABLE 1
Physical Measurements of Materials

Material	Roughness	Stiffness (R^2) (N/mm)	Friction (R^2)	Fineness (mm^{-1})
T1	0.3252	42.20 *(0.99)*	0.18 *(0.74)*	1.25
T2	0.3515	17.82 *(0.98)*	0.36 *(0.87)*	1.04
T3	0.3702	10.81 *(0.99)*	0.32 *(0.85)*	0.98
T4	0.2776	37.86 *(0.98)*	0.30 *(0.86)*	1.46
T5	0.2842	15.07 *(0.99)*	0.49 *(0.96)*	1.18
T6	0.3053	42.40 *(0.99)*	0.23 *(0.86)*	1.07

T1: Rough plastic, T2: Canvas, T3: Floor tile, T4: Silk, T5: Vinyl, T6: Wood

represents a high point (a peak) and a dark pixel represents a low point (a valley) on the texture. The presence and height of surface features (bumps) on the surface can be determined by comparing the grayscale values of adjacent pixels [33]. To estimate the roughness of the textures, which is measure of the surface feature height, we calculated the symmetric grey-level co-occurrence matrix (GLCM) for each image. The GLCM is a measure of how often combinations of pixel grayscale values occur in an image. We divided the range of grayscale values into 500 equally spaced bins and calculated the GLCM for four directions $(0°, 45°, 90°, 135°)$, which were then summed to obtain a spatially invariant GLCM for the image. The GLCM was normalized by the sum of all components to express it as a probability. Using the normalized GLCM, the roughness is estimated as the angular second moment (ASM):

$$\mathrm{ASM} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} M_c(i,j)^2, \tag{1}$$

where i is the row number, j is column number, and M_c is the normalized GLCM [33]. The estimated roughness parameters, R_a, are shown in Table 1. Although these values are not direct physical measurements of the surface, they provide an accurate representation of surface roughness and allow for the roughnesses of different materials to be directly compared.

3.4.2 Surface Stiffness Measurements

Surface stiffness was found by measuring the force-displacement curve of each material. A MTS LSB.501 50 N load cell was attached to a MTS Criterion Model 43 load frame with a position resolution of 0.06 μm. A stainless steel 1.6 mm hemispherical tooltip was attached to the load cell, which was then lowered onto the material until a load of 10 N was achieved. Position and force measurements were recorded at 10 Hz. Due to mechanical compliance in the system, only data above 1.0 N was used in the analysis. Similarly, data beyond 3.0 N was discarded because the acrylic backing strongly biased the stiffness measurements for large forces. Furthermore, 3.0 N was the maximum force typically applied by humans when interacting with a texture through a tool [10]. This range of forces is similar to that used to calculate surface compressibility in [34]. A line was fit to the force-position data between these two limits. The slope of the line is a measurement of the surface stiffness in N/mm, as shown in Table 1. No nonlinear behavior was observed, as indicated by the high R^2 values.

3.4.3 Friction Measurements

The slipperiness of the materials, which can be represented as the coefficient of kinetic friction, was found using data recorded by the haptic recording device. The experimenter dragged the tool across the surface in a fluid motion and recorded ten seconds of interaction data. The force measurements were separated into the tangential and normal components, and a line was fit to the plot using a robust fit method in which outliers were removed. The slope of the line is the kinetic friction coefficient, as shown in Table 1. The chosen materials closely follow the proposed Coulomb friction model for the forces used, as indicated by the R^2 values.

3.4.4 Fineness Measurements

The images from Section 3.4.1 were also used to calculate fineness, the materials' spatial periodicity, as the average spatial frequency of the surface features. Each row and column of pixels was transformed into the frequency domain by taking the discrete Fourier transform (DFT) of the grayscale pixel values. This procedure gave frequencies in units of pixel^{-1}, which were converted to absolute spatial frequencies in mm^{-1}. The DFT was averaged across all rows and columns, and the spectral centroid was calculated. The spectral centroid has commonly been used as a measurement of the fineness of a texture [10], [35]. The fineness parameters are shown in Table 1.

4 Texture Modeling

This section details our method for creating haptic texture models from short acceleration, force, and speed signals that are recorded as described above. First we discuss the time series model used to represent the texture data. Then we describe the segmenting method used to parse the data into short stationary segments. Finally, we describe how the models are stored for later use in rendering virtual haptic textures. Fig. 4 shows a full diagram of the texture modeling process.

4.1 Model Structure

In this work, as in past works [7], [8], [10], we use an autoregressive (AR) model structure. An AR model is an all-pole infinite impulse response (IIR) filter in which the system's next output is modeled as a linear combination of its previous outputs. The AR model order, p, specifies the number of previous output values used. The AR model structure is:

$$A(p)\ \ y(t) = e(t) + u(t), \tag{2}$$

where $A(p)$ is the array of AR coefficients, and $y(t)$, $e(t)$, and $u(t)$ are respectively the output, white-noise disturbance value, and residual at discrete time t. The discrete-time transfer function of an AR model is:

$$H(z) = \frac{1}{1 - \sum_{k=1}^{p} A_k z^{-k}}. \tag{3}$$

The coefficients of each AR model are computed by minimizing the residual error variance using the Levinson-Durbin algorithm.

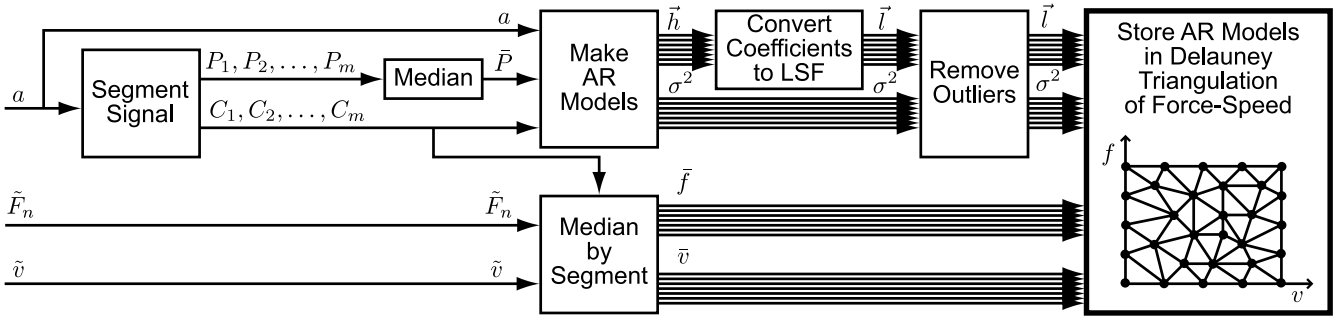

Fig. 4. A diagram of our texture modeling system. The acceleration signal is fed into the segmenting algorithm, which returns the segmentation points and proposed AR order. An AR model is created for each segment and labeled with the median force and median speed during that segment. The coefficients are converted to LSFs to ensure stability during rendering. Outlier AR models are determined and rejected as appropriate. The remaining AR models are stored in a Delaunay triangulation in force-speed space.

4.2 Data Segmentation

The recorded acceleration signals are not stationary because their power and frequency content directly depend on the unconstrained movement of the tool. However, the use of the AR model structure requires the assumption of strong stationarity. In order to make this assumption valid, we break the acceleration signal down into a sequence of approximately stationary segments to model the signal as a piecewise autoregressive process. An individual AR model is then created for each segment. For each material, we segment the recorded acceleration signal using the Auto-PARM algorithm presented in [36]. This algorithm determines the segmentation breakpoints and the AR order of each segment by optimizing the minimum description length (MDL) objective function:

$$\begin{aligned} \text{MDL} = &\log(m) + (m+1)\log(n) + \sum_{j=1}^{m+1} \log(p_j) \\ &+ \sum_{j=1}^{m+1} \frac{p_j + 2}{2} \log(n_j) + \sum_{j=1}^{m+1} \frac{n_j}{2} \log(2\pi\sigma_j^2), \end{aligned} \tag{4}$$

where m is the number of breakpoints, n is the length of data, and p_j, n_j, and σ_j^2 are respectively the AR order, length, and variance of the jth segment. The MDL function seeks to maximally compress the data by finding the best combination of the number of segments, the number of AR coefficients, and the resulting error. This maximum compression seeks to avoid the problem of oversegmenting the acceleration signal; oversegmenting may leave insufficient data to make a good AR model or may create redundant models. It also avoids the problem of overfitting individual AR models because although a model with more coefficients may fit the recorded data more precisely, it will be poor at prediction because it also fits the noise and random effects present in the data. A minimum segment span constraint of 50 datapoints was enforced to further prevent oversegmentation.

The recorded acceleration signal was fed into the Auto-PARM algorithm, which returned the location of the segment breakpoints and a proposed AR order for each segment. The number of AR coefficients was allowed to vary from one to twenty with equal probability. Auto-PARM implements an island-model genetic algorithm with 25 islands, which allows parallel computation to speed up convergence. A mutation operation prevents the algorithm from being trapped in local optima. The resulting segmentation for the acceleration signal recorded from canvas is shown in Fig. 5. Note that the segment boundaries often align with changes in either the force or speed signal even though only the acceleration signal was used in the segmentation process. For the data set shown, the algorithm resulted in 19 segments with a mean length of 5,230 datapoints (523 ms).

4.3 Creation of Texture Model Set from Segments

Once the breakpoints and AR orders for a material's data set are returned by the segmenting algorithm, a texture model is created for that material as a set of autoregressive models. All AR models for a single material must have the same number of coefficients due to the need for interpolation between the AR models, as discussed later. We choose the median of the AR orders selected for all segments for that material, which for the data shown in Fig. 5 was 19 coefficients. We refit each segment with this median AR order and determine the segment's coefficients and variance. The AR coefficients are converted to line spectral frequencies (LSF) to ensure stability during rendering, as discussed in Section 5.1.1. Each AR model is associated with the median force and speed of the segment from which it was constructed.

Although Wiertlewski et al. [37] found that textures can be represented as a function of net position when touched with

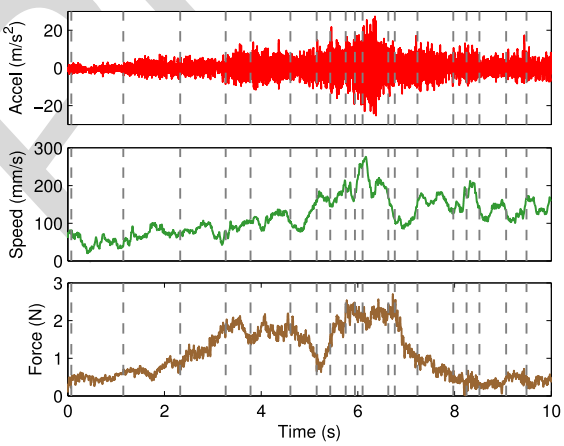

Fig. 5. Recorded acceleration that was segmented using the AutoPARM algorithm. Dotted lines represent the segment breakpoints. Force and speed were not used during the segmenting process, but they are shown to illustrate the motions the subject used during data capture.

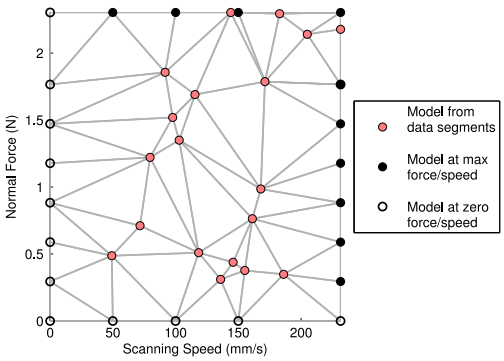

Fig. 6. Delaunay triangulation for the canvas model set. Each vertex represents an AR model of a vibration segment recorded at that combination of force and speed.

a bare finger, we found that this approach was not applicable when using a tool because the dynamics of the tool-hand-surface system drastically changes as force and speed vary. For example, the tooltip may glide across the surface features when speed is increased [38], which alters both the power and spectral content of the resulting tool vibrations. Similarly, an increase in normal force requires the human to increase grip force, which increases the rigidity of the system and changes its vibrational characteristics. Thus, this work focuses on speed and force as separate inputs to the system.

Due to the unconstrained method by which data is recorded, there is the chance that outlier AR models could exist if either force or speed varied greatly during a segment or if the tool or surface slipped during recording. Therefore, it is necessary to check for the existence of outlier AR models and remove them from the texture model set if appropriate. The AR models are placed in force-speed space and are compared to surrounding AR models on the basis of five criteria: the variance of error of the AR model and the power, spectral centroid, skewness, and kurtosis of the original segment. The expected behavior of the system is that these five values should increase monotonically as either force or speed increases. So a plane is fit to each of the five values versus force and speed. Potential outlier AR models are determined by the confidence intervals of the studentized residuals' t-distributions. If an AR model is determined to be an outlier under two or more criteria, the AR model is removed from the texture model set.

4.4 Model Storage

The segmentation and modeling process described above yields a set of AR models that are scattered in force-speed space. We thus construct a Delaunay triangulation using the z-score of the AR model forces and speeds, as shown in Fig. 6. Each vertex of a triangle represents the location of an AR model in force-speed space.

The triangulation is extended to form a rectangular convex hull by adding linearly spaced AR models at maximum modeled force, maximum modeled speed, zero force, and zero speed. This step, which is done to simplify the rendering process, is accomplished by copying the coefficients of the closest AR model. For AR models at the maximum modeled force or speed, the variance is copied as well. For AR models at zero force or zero speed, the variance is set to zero so that no vibrations are generated if the user is not touching the surface or is stationary.

5 Texture Rendering

This section describes the steps necessary to render synthetic textures using the texture model set created in Section 4. First we detail the method for generating a synthetic vibration signal, then we present the hardware necessary for rendering to a human user. Fig. 7 shows the full diagram of the texture rendering process.

5.1 Generating Vibration Signals

After the texture model sets are created, they are displayed on an updated version of the TexturePad described in [8]. This system consists of a Cintiq 12WX interactive pen display by Wacom Co., Ltd., which is capable of providing measurements of the user's force and position at a rate of 125 Hz. The Linux Wacom Project code base [39] provides access to these variables. The force sensor has a resolution of 0.0013 N, and the x-position and y-position measurements have a resolution of 4.934 and 4.861 μm, respectively.

5.1.1 AR Model Interpolation

The TexturePad system measures the user's current force and speed at 125 Hz. These force and speed values are used to determine which AR models to use in generating the vibration signal for display to the user. Simply choosing the AR model closest to the user's current force and

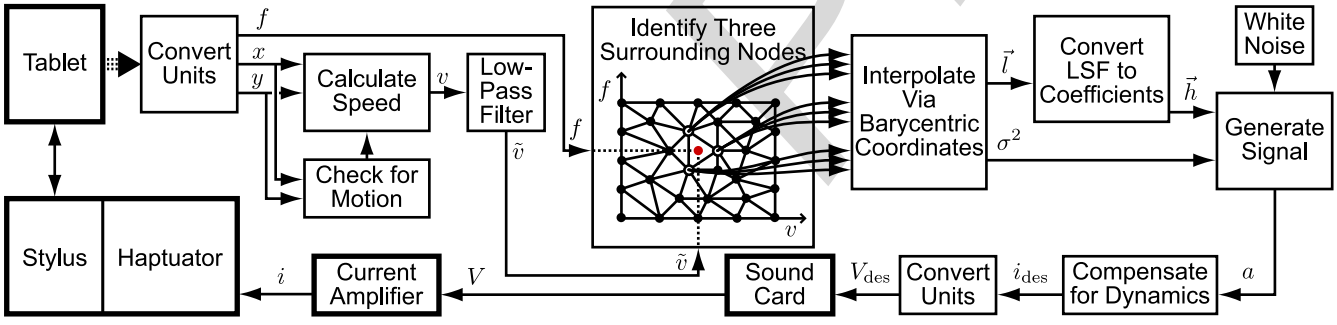

Fig. 7. A diagram of our texture rendering system. The user's current position and force are measured by the Wacom tablet. The position data is differentiated and low-pass filtered to obtain the speed. A collision detection algorithm is run to determine where the user's force and speed lie in the Delaunay triangulation. Barycentric coordinates are used to interpolate between the LSFs and variance of the three AR models at the vertices of the triangle. The interpolated LSFs are transformed to AR coefficients. The coefficient and variance values are passed to our LPC signal generator, which creates an appropriate acceleration waveform. This signal is output via the computer soundcard, through a current amplifier, to drive the Haptuator on the stylus.

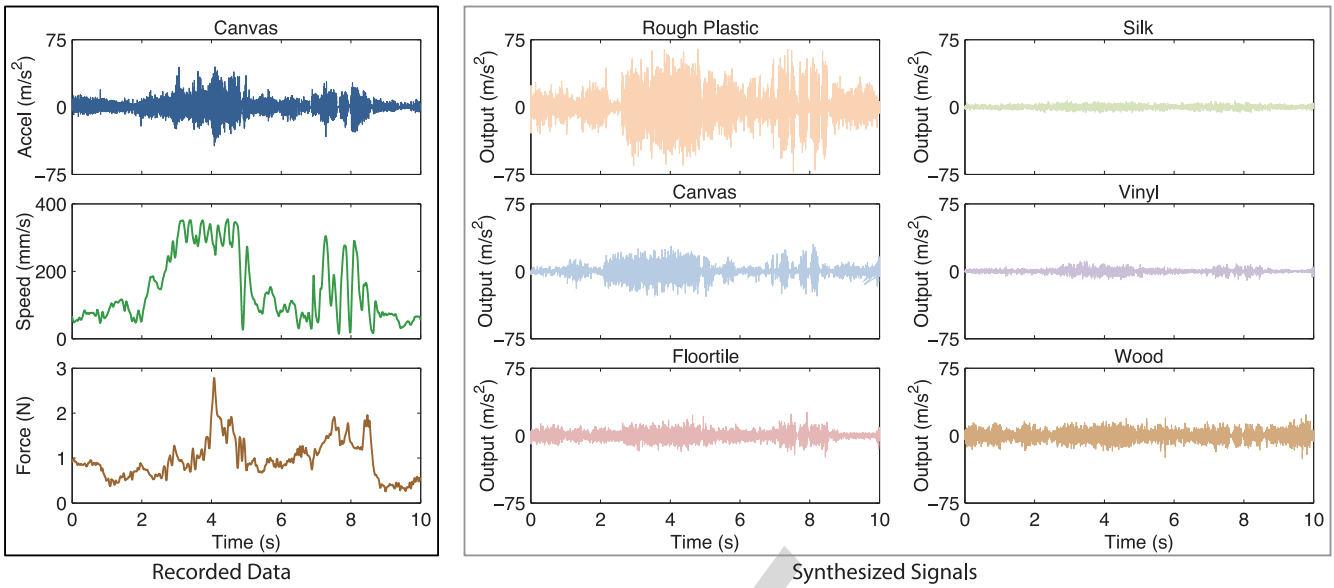

Fig. 8. Recorded and synthesized data for comparison of texture model set performance. The acceleration, force, and speed data were recorded from the canvas material using the haptic recording device. The recorded force and speed signals were then used to synthesize accelerations for all six materials.

speed creates the potential for high-frequency model switching, which can cause instability [40]. Similarly, directly interpolating the AR models' coefficients can move the poles of the resulting transfer function outside the unit circle, making the transfer function unstable. Therefore, to ensure a stable system, one must convert each AR model's coefficients into line spectral frequencies before interpolation, as done in [9], [10]. The LSFs are found by mapping the AR model's poles in the discrete plane onto the unit circle and are defined as the angles these mapped poles make with the real axis. This pole mapping ensures the poles remain inside the unit circle, which is a necessary condition for stability.

We use a visibility-walk collision-detection algorithm [41] on the Delaunay triangulation to determine which triangle contains the user's current force and speed point. The user's force and speed are saturated at the maximum modeled force and maximum modeled speed to ensure that the force-speed trajectory remains inside the convex hull of the triangulation. Barycentric coordinates are used to provide a weighted average of the LSFs and variance values of the three AR models at the triangle vertices. The interpolated LSFs are then converted back into AR coefficients. New coefficients and variance values are calculated at a rate of 125 Hz.

5.1.2 *Signal Synthesis*

A white Gaussian excitation signal is generated with power equal to the interpolated variance value. Histories of the excitation signal and vibration outputs are used to calculate the new vibration output at a rate of 10 kHz.

Fig. 8 shows synthesized texture vibration signals for the six chosen materials. Speed and force signals recorded while touching real canvas were used to synthesize acceleration signals for all six textures to show their varied output responses. The signal synthesized using the canvas model set has the best match to the recorded acceleration signal.

5.2 Rendering Vibration Signals

5.2.1 *Dynamic Compensation*

After the next acceleration value is synthesized and before it is played to the user, a feed-forward dynamic compensation controller is implemented to diminish the resonance of the Haptuator and achieve a higher-fidelity recreation of the tool vibrations. This dynamic compensation scheme, which is outlined in [42], approximately inverts the system's dynamics to estimate the current output needed to create the desired tool vibration.

5.2.2 *Hardware*

The synthesized acceleration signal is output through the soundcard of the computer using the PortAudio C library [43]. This output voltage is then passed through a linear current amplifier with a gain of 1 A/V. The synthesized vibrations are transmitted to the user by driving a Haptuator vibrotactile transducer (TactileLabs, model no. TL002-14-A) that is firmly attached to the stylus via a custom 3D-printed plastic bracket.

6 Experimental Methods

This section describes the human-subject experiment we ran to evaluate the realism of our virtual textures and the strengths and weaknesses of this approach. In particular, we wanted to determine the extent to which our modeling and rendering approach captures the material properties shown in Table 1. To accomplish this, we adapted the study and analysis conducted in [13]. All procedures were approved by the Penn IRB under protocol 816548, and subjects gave informed consent. Two separate sets of ten subjects were recruited for a total of twenty subjects. The subjects were all right-handed with limited experience with haptic devices.

6.1 Set-up

The subject sat at a table in front of the tablet and across from the experimenter. As shown in Fig. 9, a black curtain

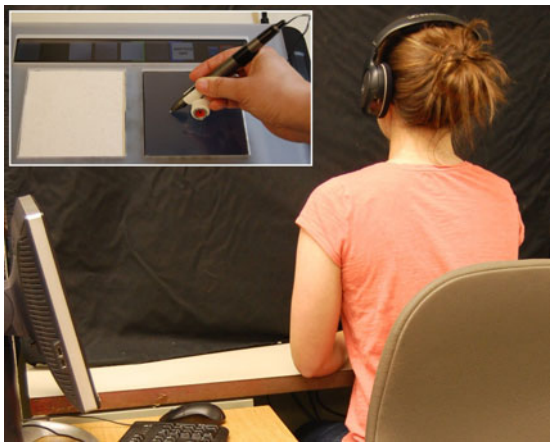

Fig. 9. Experimental set-up. The subject sat in front of the tablet, across from the experimenter. A black curtain prevented the subject from seeing the textures, and headphones playing pink noise masked audio cues. The subject grasped the stylus augmented with a Haptuator in her dominant hand and interacted with real and virtual textures placed on the tablet screen.

was hung between the subject and the tablet, and the subject wore headphones playing pink noise. These interventions ensured that the subject used only haptic, and not visual and auditory, cues during the study. The subject used the augmented stylus to interact with the real and virtual textures. The tablet measured the user's force and speed during all texture interactions. When a virtual texture was present, the force and speed values were used to render the texture vibration signal, as described in Section 5. When a real texture was present, the Haptuator was not activated. An ADXL345 digital accelerometer configured into ± 78.4 m/s^2 (± 8 g) mode with a resolution of ± 0.153 m/s^2 was rigidly attached to the stylus and polled at 800 Hz to record the tool vibrations. Subjects were instructed to keep the stylus on the surface at all times except when switching between textures and were not allowed to tap or statically press the stylus into the surface. The experimenter monitored the subject's hand to ensure that it did not come in contact with the texture or tablet. A 0.6 cm thick acrylic piece with two 10.2 cm square cutouts was mounted to the face of the tablet. The cutouts were used to secure the materials during the study and to act as a guide to keep the stylus constrained to the textures. For virtual textures, a 0.32 cm thick piece of acrylic was placed in the cutout on the tablet screen so thickness cues could not be used to determine the presence of a virtual texture.

6.2 Procedure

There were three separate phases to the study: free exploration, pairwise comparisons, and adjective ratings. One set of ten subjects completed the pairwise comparison and adjective ratings phases of the study, but that adjective rating data was not used in this analysis because order effects may have been present. A second set of ten subjects completed the free exploration and adjective rating phases. The study used the real and virtual versions of five of the textures shown in Fig. 3, for a total of ten textures (five real, five virtual). Wood was used only in the pairwise comparison phase of the study, but the data associated with the real and virtual versions of wood were not used in the analysis due to its anisotropy [10]. Study data were collected and managed using Research Electronic Data Capture (REDCap) tools hosted at the University of Pennsylvania [44].

6.2.1 Free Exploration

In this phase of the study, the ten textures were presented to the subject one at a time in randomized order. At the beginning of each trial, the experimenter placed one texture in the right cutout on the tablet and then turned on an LED to signal that the texture was ready. The subject placed his or her hand under the curtain and was given the stylus, which he or she used to explore each texture for as long as desired. The first ten seconds of acceleration, force, and speed interaction data, as measured by the tablet, were recorded for each texture. The subject then removed his or her hand from under the curtain and typed a short description of that texture on a nearby computer. This procedure was repeated until the subject had felt all ten textures. This phase allowed the subjects to become familiar with the textures and the experimental set-up.

6.2.2 Pairwise Comparisons

In this study phase, all possible pairs of textures (real-real, virtual-virtual, and real-virtual) were presented to the subject one at a time for a total of 45 pairs. The textures were presented in randomized order with equal likelihood of being placed in the left or right cutout. For each pair, the experimenter placed the two textures in the cutouts on the screen and turned on an LED to signal that the textures were ready. The subject felt both textures with the stylus, switching between the two as much as desired. Subjects then rated the similarity of the two textures by placing a mark on a scale of "completely the same" to "completely different" on the computer using a mouse wielded in their non-dominant hand.

6.2.3 Adjective Ratings

In the final phase of the study, the textures were presented to the subject one at a time in randomized order. In addition to the ten textures described above, the subjects were also presented with an acrylic piece without a virtual texture present. The subject was asked to rate each texture on the scales of rough-smooth, hard-soft, slippery-not slippery, and fine-coarse. These scales were chosen from the list provided in [45] as the psychophysical dimensions applicable to the perception of textures through a tool. The subject followed the same procedure as above but felt only the texture in the right cutout. The experimenter did not define the adjectives, so the subject had to rely on his or her intuitive understanding of the scales. The subject rated all textures along each scale before proceeding to the next scale.

7 Results

7.1 Free Exploration Recordings

The acceleration recordings from the free exploration phase of the study were used to compare the power and spectral centroid of the accelerations experienced when interacting with real and virtual versions of the same texture. The force

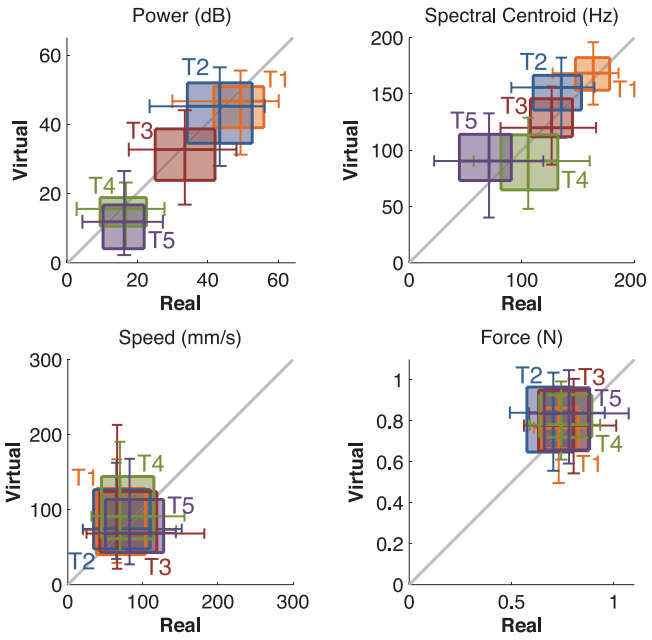

Fig. 10. Comparison of the power and spectral centroid of the recorded acceleration signal and the speeds and forces used by subjects during the free exploration phase of the study. There is a strong linear trend for power and spectral centroid comparing values for the corresponding real and virtual textures. There is no statistical difference in the speeds and forces used by subjects on real and virtual textures.

and speed recordings were also compared to determine if there was any difference in the speeds and forces commonly used by subjects touching real and virtual textures. The mean and one standard deviation above and below the mean of the acceleration power, acceleration spectral centroid, speed, and force for each ten-second-long recording were found. These values were combined across all subjects for each texture. The resulting two-dimensional box plots are shown in Fig. 10, allowing for direct visual comparison of the recorded values for corresponding real and virtual textures. The central lines of each box denote the real and virtual medians, the edges of each box denote the 25th and 75th percentiles, and the whiskers extend to the extreme datapoints not considered outliers. A line denoting the ideal relationship between the real and virtual ratings is shown. Two-way analysis of variance (ANOVA) was performed separately on the power, spectral centroid, speed, and force values using texture and virtual/real as factors. The differences in the power of the acceleration signals were found to be statistically significant across different textures ($F = 101.2$, $p < 0.001$), but not statistically significant when comparing real and virtual textures ($F = 2.48$, $p = 0.12$). Similarly, the differences in the spectral centroid of the acceleration signals were found to be statistically significant across different textures ($F = 72.52$, $p < 0.001$), but not statistically significant when comparing real and virtual textures ($F = 2.31$, $p = 0.13$). No statistically significant differences were found in the speeds used by the subjects across different textures ($F = 0.48$, $p = 0.75$) and comparing real and virtual textures ($F = 1.48$, $p = 0.22$). Similarly no statistically significant differences were found in the forces across different textures ($F = 1.19$, $p = 0.32$) and comparing real and virtual textures ($F = 2.33$, $p = 0.13$). The free response descriptions provided by the

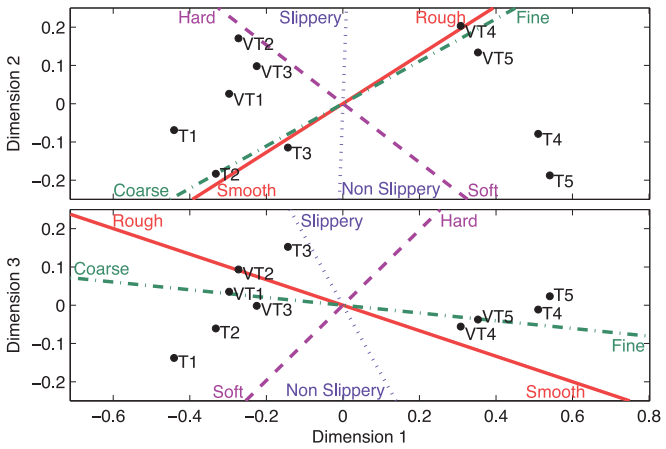

Fig. 11. Real and virtual textures placed in three-dimensional perceptual space based on dissimilarity ratings. Lines showing the perceptual qualities of roughness, hardness, slipperiness, and fineness were fit to the space using ratings along the four adjective scales.

subjects during this phase were not analyzed because their purpose was to allow subjects to form an unbiased perception of the textures before being asked to rate them along the four adjective scales.

7.2 Pairwise Comparisons

Each subject's pairwise dissimilarity matrix was normalized by the standard deviation of their dissimilarity ratings to compensate for individual differences in subjective rating scales. The normalized dissimilarity matrices were averaged across the ten subjects. Multi-dimensional scaling (MDS), which is a common tool in haptic texture research for visualizing perceived differences between materials [13], [34], was used on the resulting average dissimilarity matrix to place the ten textures in an N-dimensional perceptual space. The distance between a pair of textures in the perceptual space is proportional to their average rated dissimilarity. An analysis of the scree plot for up to six dimensions indicated that two or three dimensions were necessary to achieve a good fit ($R^2 > 0.9$). Fig. 11 shows the resulting fit of the ten textures into a three-dimensional perceptual space.

Fig. 12 shows the normalized dissimilarity ratings before averaging, separated into the four types of texture pairs (real-virtual same texture, real-real different textures, real-virtual different textures, virtual-virtual different textures). The median dissimilarity rating for the real-virtual same texture pairs is much lower than the median ratings for the

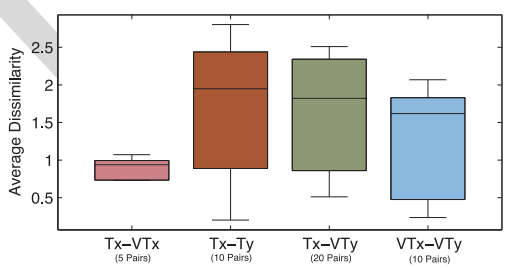

Fig. 12. Comparison of dissimilarity ratings across the four types of texture pairs. The median dissimilarity rating for the real-virtual same texture pairs is significantly lower than the median ratings for the three other types of pairs.

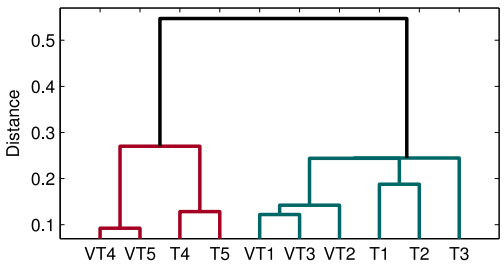

Fig. 13. Dendrogram of materials showing clustering of real and virtual textures using the distances between textures in three-dimensional perceptual space.

three other types of pairs, indicating that the corresponding real and virtual texture pairs were rated as more similar than pairs of different textures.

In order to understand how the textures were distributed in the perceptual space, a cluster analysis was conducted on the Euclidean distances between textures. A hierarchical binary cluster tree was generated, and the resulting dendrogram is shown in Fig. 13.

7.3 Adjective Ratings

The adjective ratings were separately normalized for each subject along the four scales by dividing by the standard deviation along that scale for that subject. The normalized adjective ratings were then averaged across all subjects. The normalized adjective ratings before averaging are shown in the two-dimensional box plots in Fig. 14. A two-factor ANOVA was performed separately on the four sets of adjective ratings using texture and virtual/real as factors. When comparing across textures, the differences in ratings of roughness ($F = 52.18$, $p < 0.001$), hardness ($F = 24.46$, $p < 0.001$), and fineness ($F = 44.76$, $p < 0.001$) were found to be

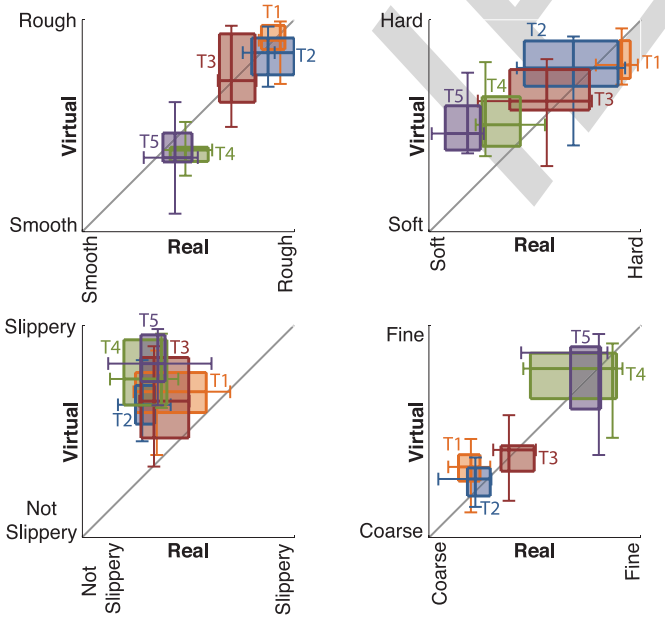

Fig. 14. Comparison on subjects' ratings of the ten textures along the four adjective scales. The box plots for roughness and fineness show a strong linear trend in the ratings for the corresponding real and virtual textures. The box plot for hardness shows a similar trend in the ratings, while the box plot for slipperines shows no correlation.

TABLE 2
Correlation of Adjective Lines

Adjective	Roughness	Hardness	Slipperiness	Fineness
Roughness	1.00	0.81	−0.25	−0.98
Hardness	0.81	1.00	0.22	−0.83
Slipperiness	−0.25	0.22	1.00	0.24
Fineness	−0.98	−0.83	0.24	1.00

statistically significant. The differences in slipperiness ratings were not found to be statistically significant across textures ($F = 1.74$, $p = 0.15$). When comparing virtual and real textures, the differences in hardness ($F = 9.72$, $p < 0.005$) and slipperiness ($F = 99.91$, $p < 0.001$) ratings were found to be statistically significant. The differences in roughness ($F = 2.42$, $p = 0.12$) and fineness ($F = 2.37$, $p = 0.13$) ratings were not found to be statistically significant when comparing virtual and real textures.

We separately regressed the four sets of average adjective ratings onto the perceptual space to place the four adjective lines, as shown in Fig. 11. In the regression, each set of adjective ratings was used as the dependent variable with the coordinates of the textures in the perceptual space as predictor variables. The independence of the adjective lines was examined by analyzing the correlations in Table 2. There was a high degree of negative correlation between the roughness and fineness scales, and between fineness and hardness ($p < 0.005$). There was also a high degree of positive correlation between roughness and hardness ($p < 0.005$). No other correlations were statistically significant.

To determine the correlation between perceived dissimilarity of the textures and the ratings along the four adjective scales, we performed a multilinear regression on the average dissimilarity ratings. The pairwise differences in the average ratings of textures along the four adjective scales were used to predict dissimilarity, $\hat{D}$:

$$\hat{D} = b_1 d_r + b_2 d_h + b_3 d_s + b_3 d_f, \tag{5}$$

where b_1, b_2, b_3, and b_4 are the regression coefficients, and d_r, d_h, d_s, and d_f are the respective differences in the roughness, hardness, slipperiness, and fineness between textures. Fig. 15 shows the normalized perceived versus predicted dissimilarity. There was found to be a high degree of correlation ($R^2 = 0.88$) between the predicted and perceived dissimilarity values. Therefore, approximately 88 percent of

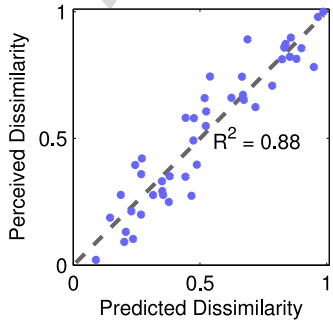

Fig. 15. Normalized perceived versus predicted dissimilarity. The perceived dissimilarity is found using the dissimilarity ratings averaged across all subjects. The predicted dissimilarity is found by performing multilinear regression on the average dissimilarity ratings using as predictors the pair-wise differences in the average ratings of textures along the four adjective scales.

the dissimilarity between the textures can be explained by the differences in the perceived roughness, hardness, slipperiness, and fineness of the textures.

Comparing the adjective ratings to the measured parameters in Table 1, the subjects performed best at discriminating the differences in roughness of the real textures ($R^2 = 0.54$) and marginally worse at discriminating differences in slipperiness ($R^2 = 0.31$) and fineness ($R^2 = 0.29$) although none of these correlations are statistically significant. The rated hardness of the textures shows the greatest difference from the relative measured hardness ($R^2 = 0.13$).

8 Discussion

By evaluating how humans perceive both real and virtual textures, the study provided many important insights into the strengths and weaknesses of our modeling and rendering techniques. Since it was shown that the speeds and forces used by subjects were not statistically different across textures, the perceived dissimilarities and the corresponding adjective ratings were most likely not influenced by different interaction conditions. The power and spectral centroid of the resulting vibrations, however, were statistically different across textures and were presumably used by subjects during rating.

The dendrogram in Fig. 13 shows that there are two distinct clusters separating virtual and real silk and vinyl from virtual and real rough plastic, canvas, and floor tile. Comparing this clustering to the adjective ratings in Fig. 14 and the measured physical parameters in Table 1, one can conclude that the main distinction between the two groups is their respective roughnesses. Furthermore, from the ANOVA analysis on the adjective ratings along the four scales, we determined that the distinction between virtual and real textures is not statistically significant when comparing the rated roughness of the surfaces. Thus, our modeling and rendering methods accurately capture and recreate the perceptual feeling of roughness of real textures, which supports the use of tool vibrations to represent roughness [13], [46], [47], [48]. However, textures that were rated rough were also commonly rated coarse, and textures rated smooth were also rated fine. Although roughness and fineness were used as separate perceptual dimensions in previous research [35], the high level of correlation between the two reveals that subjects were not able to effectively separate the dimensions in this study. This interaction is supported by the study ran in [49], which indicated that both physical roughness and fineness contribute to a subject's personal definition of perceived roughness.

Although the clustering of the textures along the rough/smooth parameter shows promise for our methods, the real and virtual versions of the same textures are not directly clustered together. This disparity indicates that other perceptual parameters that are not being directly modeled and rendered are important in the realism of virtual textures. Differences in the rated hardness of textures were significant between real and virtual surfaces, but Fig. 14 does show similar trends across the five textures of both types. Since all of the virtual textures were rendered on the same piece of acrylic, and since hardness was not directly modeled or rendered, one might expect all of the virtual textures to be rated the same hardness as the acrylic piece, which was hardest of the materials (52.1 N/mm). Since the virtual textures received different hardness ratings, the vibrations that were played to the subject seem to have created an illusion of differing hardnesses, which we would expect because of the interplay between the vibrational characteristics of the tool-hand-surface system and the surface stiffness. More tests would be needed to confirm the effect of data-driven texture vibrations on the perceived hardness of a surface. However, in order to fully capture the hardness dimension, the surface stiffness must be rendered separately using a force-feedback device. Additionally, the hardness ratings of the real textures showed the lowest correlation to the measured stiffness parameters among the four parameters. This inconsistency may be due to the fact that subjects were not allowed to tap or statically press into the surface, which are the motions humans typically use to evaluate the hardness of an object [50].

Further variation in the clustering can be explained by the slipperiness ratings. Each virtual texture was consistently rated as more slippery than its corresponding real texture, as indicated in Fig. 14. However, there was not significant variation in the rated slipperiness when comparing across textures for either real or virtual textures. This similarity is not supported by the measured coefficients of friction in Table 1, indicating that there were factors other than friction affecting the perception of slipperiness of the real textures. When a tool is dragged across a real rough surface, the tooltip catches on the bumps, which cause impulses to the tooltip in addition to the friction of the material. Thus, it may have been difficult for subjects to separate the slipperiness of a real texture from its roughness [48], and the higher slipperiness ratings of the virtual textures may be due partly to their inability to provide these hard stops. This discrepancy in slipperiness may also be partly due to the choice of the acrylic piece as the base on which to render the virtual textures. The acrylic piece was rated as the most slippery, which agrees with its measured coefficient of friction ($\mu_k = 0.13$). Choosing a less slippery base material would potentially increase the slipperiness match between real and virtual by decreasing the slipperiness of the virtual textures, but fully capturing slipperiness requires that friction be rendered directly using a force-feedback device.

9 Conclusion

This paper presented methods for modeling and rendering haptic virtual textures from data recorded during unconstrained tool-surface interactions. First we presented our data-recording hardware, which measures the speed, force, and high-frequency acceleration of the tool as a user drags it across a texture. Since the user's speed and force vary during data capture, the acceleration signal is not stationary. Therefore, we segment the signal to represent it as a piecewise autoregressive process. This procedure results in a texture model set that represents the feel of the texture at many input conditions. Each texture model set is stored in a Delaunay triangulation with each vertex representing a single model. We then presented a method for using these texture model sets to create realistic virtual textures in real

time. The TexturePad system measures a user's force and speed and outputs a spectrally accurate time-domain vibration signal to a Haptuator.

We ran a study that evaluated the realism of our virtual textures and the strengths and weaknesses of our modeling approach. Subjects rated the similarity of pairs of textures. Subjects also rated all textures along four adjective scales (roughness, hardness, slipperiness, fineness). The study indicated that our virtual textures accurately capture and recreate the roughness of the real textures. However, the rated hardness and slipperiness of the virtual textures did not correlate well with the real textures. Therefore, hardness and slipperiness should be added to the rendering process to improve the realism of our virtual textures. Our current TexturePad system is incapable of modulating stiffness or friction, so a different rendering platform must be implemented to provide these other aspects of the textures. We are currently working toward using more traditional haptic devices to render our textures [32]. This expansion will use open-source software, which will have the added benefit of allowing us to share textures with other researchers.

ACKNOWLEDGMENTS

The authors thank Joe Romano for creating the initial TexturePad, Tim Herrman and Ben Goodman for helping create the haptic recording device, and Kevin Turner for the use of his lab and equipment for the stiffness measurements. They also thank Richard A. Davis for providing the executable to run the Auto-PARM algorithm. This work was supported by the National Science Foundation under Grant No. 0845670. The work of Heather Culbertson was supported by a research fellowship from the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-0822.

REFERENCES

[1] S. J. Lederman, "Tactile roughness of grooved surfaces: The touching process and effects of macro- and micro-surface structure," *Perception Psychophys.*, vol. 16, no. 2, pp. 385–395, 1974.

[2] R. L. Klatzky, S. J. Lederman, C. Hamilton, M. Grindley, and R. H. Swendsen, "Feeling textures through a probe: Effects of probe and surface geometry and exploratory factors," *Perception Psychophys.*, vol. 65, no. 4, pp. 613–631, 2003.

[3] G. Campion and V. Hayward, "Fundamental limits in the rendering of virtual haptic textures," in *Proc. IEEE Eurohaptics Conf.*, 2005, pp. 263–270.

[4] K. J. Kuchenbecker, J. P. Fiene, and G. Niemeyer, "Improving contact realism through event-based haptic feedback," *IEEE Trans. Vis. Comput. Graph.*, vol. 12, no. 2, pp. 219–230, March/April 2006.

[5] M. A. Otaduy and M. C. Lin, "Rendering of textured objects," in *Haptic Rendering: Foundations, Algorithms, and Applications*, M. Lin and M. Otaduy, Eds., Natick, MA, USA: A. K. Peters, 2008, ch. 18, pp. 371–393,.

[6] A. M. Okamura, K. J. Kuchenbecker, and M. Mahvash, "Measurement-based modeling for haptic display," in *Haptic Rendering: Foundations, Algorithms, and Applications*, M. Lin and M. Otaduy, Eds., Natick, MA, USA: A. K. Peters, 2008, ch. 21, pp. 443–467.

[7] J. M. Romano, T. Yoshioka, and K. J. Kuchenbecker, "Automatic filter design for synthesis of haptic textures from recorded acceleration data," in *Proc. IEEE Int. Conf. Robot. Autom.*, May. 2010, pp. 1815–1821.

[8] J. M. Romano and K. J. Kuchenbecker, "Creating realistic virtual textures from contact acceleration data," *IEEE Trans. Haptics*, vol. 5, no. 2, pp. 109–119, Apr.–Jun. 2012.

[9] H. Culbertson, J. M. Romano, P. Castillo, M. Mintz, and K. J. Kuchenbecker, "Refined methods for creating realistic haptic virtual textures from tool-mediated contact acceleration data," in *Proc. IEEE Haptics Symp.*, Mar. 2012, pp. 385–391.

[10] H. Culbertson, J. Unwin, B. E. Goodman, and K. J. Kuchenbecker, "Generating haptic texture models from unconstrained tool-surface interactions," in *Proc. IEEE World Haptics Conf.*, Apr. 2013, pp. 295–300.

[11] M. Hollins, R. Faldowski, S. Rao, and F. Young, "Perceptual dimensions of tactile surface texture: A multidimensional scaling analysis," *Perception, Psychophys.*, vol. 54, no. 6, pp. 697–705, 1993.

[12] D. Picard, C. Dacremont, D. Valentin, and A. Giboreau, "Perceptual dimensions of tactile textures," *Acta Psychol.*, vol. 114, no. 2, pp. 165–184, 2003.

[13] T. Yoshioka, S. J. Bensmaia, J. C. Craig, and S. S. Hsiao, "Texture perception through direct and indirect touch: An analysis of perceptual space for tactile textures in two modes of exploration," *Somatosensory Motor Res.*, vol. 24, nos. 1–2, pp. 53–70, 2007.

[14] S. Saga and R. Raskar, "Simultaneous geometry and texture display based on lateral force for touchscreen," in *Proc. SIGGRAPH Asia 2012 Emerging Technol.*, 2012, p. 8.

[15] C. McDonald and K. J. Kuchenbecker, "Dynamic simulation of tool-mediated texture interaction," in *Proc. IEEE World Haptics Conf.*, Apr. 2013, pp. 307–312.

[16] M. Minsky, O. Y. Ming, O. Steele, J. F. P. Brooks, and M. Behensky, "Feeling and seeing: Issues in force display," in *Proc. Symp. Interact. 3D Graph.*, 1990, vol. 24, pp. 235–241.

[17] J. Fritz and K. Barner, "Stochastic models for haptic texture," in *Proc. SPIE Int. Symp. Intell. Syst. Adv. Manuf.*, 1996, pp. 34–44.

[18] C. Basdogan, C. H. Ho, and M. A. Srinivasan, "A ray-based haptic rendering technique for displaying shape and texture of 3D objects in virtual environments," in *Proc. ASME Dyn. Syst. Control Div.*, 1997, pp. 77–84.

[19] A. M. Okamura, R. J. Webster III, J. T. Nolin, K. Johnson, and H. Jafry, "The haptic scissors: Cutting in virtual environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1, 2003, pp. 828–833.

[20] M. Harders, G. Bianchi, B. Knoerlein, and G. Székely, "Calibration, registration, and synchronization for high precision augmented reality haptics," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 1, pp. 138–149, Jan/Feb. 2009.

[21] S. Jeon, S. Choi, and M. Harders, "Rendering virtual tumors in real tissue mock-ups using haptic augmented reality," *IEEE Trans. Haptics*, vol. 5, no. 1, pp. 77–84, Jan.–Mar. 2012.

[22] A. M. Okamura, J. T. Dennerlein, and R. D. Howe, "Vibration feedback models for virtual environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1998, pp. 674–679.

[23] V. L. Guruswamy, J. Lang, and W. S. Lee, "Modeling of haptic vibration textures with infinite-impulse-response filters," in *Proc. IEEE Int. Workshop Haptic Audio Vis. Environ. Games*, Nov. 2009, pp. 105–110.

[24] Y. Takeuchi, S. Kamuro, K. Minamizawa, and S. Tachi, "Haptic duplicator," in *Proc. ACM Virtual Reality Int. Conf.*, 2012. Q1

[25] D. K. Pai, K. v. d. Doel, D. L. James, J. Lang, J. E. Lloyd, J. L. Richmond, and S. H. Yau, "Scanning physical interaction behavior of 3D objects," in *Proc. 28th Annu. Conf. Comput. Graph. and Interact. Tech.*, 2001, pp. 87–96.

[26] H. Vasudevan and M. Manivannan, "Recordable haptic textures," in *Proc. IEEE Int. Workshop Haptic Audio Vis. Environ.*, 2006, pp. 130–133.

[27] J. Bell, S. Bolanowski, and M. Holmes, "The structure and function of Pacinian corpuscles: A review," *Progress Neurobiol.*, vol. 42, no. 1, pp. 79–128, 1994.

[28] J. M. Yau, J. B. Olenczak, J. F. Dammann, and S. J. Bensmaia, "Temporal frequency channels are linked across audition and touch," *Current Biol.*, vol. 19, no. 7, pp. 561–566, 2009.

[29] S. Cholewiak, K. Kim, H. Tan, and B. Adelstein, "A frequency-domain analysis of haptic gratings," *IEEE Trans. Haptics*, vol. 3, no. 1, pp. 3–14, Jan.–Mar. 2010.

[30] N. Landin, J. M. Romano, W. McMahan, and K. J. Kuchenbecker, "Dimensional reduction of high-frequency accelerations for haptic rendering," in *Haptics: Generating and Perceiving Tangible Sensations*, A. Kappers, J. van Erp, W. Bergmann Tiest, and F. van der Helm, Eds., vol. 6192, Berlin, Germany:Springer, Jul. 2010, pp. 79–86.

[31] H. Culbertson, C. G. McDonald, B. E. Goodman, and K. J. Kuchenbecker, "Data-driven modeling and rendering of isotropic textures," presented at IEEE World Haptics Conference, Seoul, Korea, Apr. 2013.
[32] H. Culbertson, J. J. López Delgado, and K. J. Kuchenbecker, "One hundred data-driven haptic texture models and open-source methods for rendering on 3d objects," in *Proc. IEEE Haptics Symp.*, Feb. 2014, pp. 319–325.
[33] E. Gadelmawla, "Estimation of surface roughness for turning operations using image texture features," *Proc. Inst. of Mech. Eng., Part B: J. Eng. Manuf.*, vol. 225, no. 8, pp. 1281–1292, 2011.
[34] W. M. Bergmann Tiest and A. M. Kappers, "Analysis of haptic perception of materials by multidimensional scaling and physical measurements of roughness and compressibility," *Acta Psychol.*, vol. 121, no. 1, pp. 1–20, 2006.
[35] J. A. Fishel and G. E. Loeb, "Bayesian exploration for intelligent identification of textures," *Front. Neurorobot.*, vol. 6, 2012.
[36] R. Davis, T. C. M. Lee, and G. Rodriguez-Yam, "Structural break estimation for nonstationary time series models," *J. Amer. Statist. Assoc.*, vol. 101, no. 473, pp. 223–239, 2006.
[37] M. Wiertlewski, J. Lozada, and V. Hayward, "The spatial spectrum of tangential skin displacement can encode tactual texture," *IEEE Trans. Robot.*, vol. 27, no. 3, pp. 461–472, Jun. 2011.
[38] S. J. Lederman, R. L. Klatzky, C. L. Hamilton, and G. I. Ramsay, "Perceiving roughness via a rigid probe: Psychophysical effects of exploration speed and mode of touch," *Haptics-E*, vol. 1, no. 1, 1999.
[39] "The Linux Wacom Project," (2010, Apr.). [Online] Available: http://linuxwacom.sourceforge.net/.
[40] D. Liberzon, *Switching in Systems and Control*. Berlin, Germany: Springer, 2003.
[41] O. Devillers, S. Pion, and M. Teillaud, "Walking in a triangulation," in *Proc. ACM 17th Annu. Symp. Comput. Geom.*, 2001, pp. 106–114.
[42] W. McMahan and K. J. Kuchenbecker, "Dynamic modeling and control of voice-coil actuators for high-fidelity display of haptic vibrations," in *Proc. IEEE Haptics Symp.*, Feb. 2014, pp. 1115–122.
[43] (2010 Apr.). PortAudio—Cross-Platform Audio API. [Online]. Available: http://www.portaudio.com/.
[44] P. A. Harris, R. Taylor, R. Thielke, J. Payne, N. Gonzalez, and J. G. Conde, "Research electronic data capture (redcap)—A metadata-driven methodology and workflow process for providing translational research informatics support," *J. Biomed. Inf.*, vol. 42, no. 2, pp. 377–381, 2009.
[45] S. Okamoto, H. Nagano, and Y. Yamada, "Psychophysical dimensions of tactile perception of textures," *IEEE Trans. Haptics*, vol. 6, no. 1, pp. 81–93, First Quarter 2012.
[46] R. L. Klatzky and S. J. Lederman, "Tactile roughness perception with a rigid link interposed between skin and surface," *Perception Psychophys.*, vol. 61, no. 4, pp. 591–607, 1999.
[47] S. J. Bensmaïa and M. Hollins, "The vibrations of texture," *Somatosens. Motor Res.*, vol. 20, no. 1, pp. 33–43, 2003.
[48] W. M. Bergmann Tiest, "Tactual perception of material properties," *Vis. Res.*, vol. 50, no. 24, pp. 2775–2782, 2010.
[49] W. M. Bergmann Tiest and A. M. Kappers, "Haptic and visual perception of roughness," *Acta Psychol.*, vol. 124, no. 2, pp. 177–189, 2007.
[50] R. H. LaMotte, "Softness discrimination with a tool," *J. Neurophysiol.*, vol. 83, no. 4, pp. 1777–1786, 2000.

Heather Culbertson received the BS degree in mechanical engineering from the University of Nevada, Reno, in 2010, and the Master's degree from the Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, May 2013, where she is currently working toward the PhD degree. She works in the Haptics Group, a part of the GRASP Laboratory. She is a student member of the IEEE.

Juliette Unwin is working toward the MEng degree in mechanical engineering at the University of Southampton, England. During Summer 2012, she was a visiting research assistant in the Haptics Group at the University of Pennsylvania, and was awarded a Royal Academy of Engineering/Nuffield Foundation Bursary to undertake research on modelling flow in porous rocks at the University of Cambridge in Summer 2013.

Katherine J. Kuchenbecker received the BS, MS, and PhD degrees in mechanical engineering from Stanford University in 2000, 2002, and 2006, respectively. She was a postdoctoral researcher at the Johns Hopkins University from 2006 to 2007. She is currently an Associate Professor in the Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania. Her research interests include the design and control of haptic interfaces, and she directs the Penn Haptics Group, which is part of the GRASP Laboratory. She serves on the program committee for the IEEE Haptics Symposium and other conferences in the fields of haptics and robotics, and has received several awards for her research, including an US NSF CAREER Award in 2009, inclusion in the Popular Science Brilliant 10 in 2010, and the IEEE Robotics and Automation Society Academic Early Career Award in 2012. She is a member of the IEEE.

Query to the Author

Q1. Please provide the page range in Refs. [24] and [35].