Haptic Texturing - A Stochastic Approach *

Juhani Siira and Dinesh K. Pai
Departments of Electrical Engineering & Computer Science
University of British Columbia
Vancouver, B.C. V6T 1Z4, Canada
e-mail: {`jsiira|pai`}`@cs.ubc.ca`

Abstract

All objects have a surface roughness which manifests itself as small forces when objects slide under load against each other. Simulating this roughness haptically enriches the interaction between a user and a virtual world, just as creating graphical textures enhances the depiction of a scene. As with graphical textures, a major design constraint for haptic textures is the generation of a sufficiently "realistic" texture given hard constraints on computational costs. We present a simple, fast algorithm to synthesize haptic textures from statistical properties of surfaces. The synthesized texture can be overlaid on other contact models, such as hard contact with Coulomb friction. The algorithm requires minimal hardware support, and can be implemented on a variety of force-feedback mechanisms. It has been successfully implemented on a two-degree-of-freedom haptic interface (the Pantograph).

1 Introduction

Haptic interfaces, such as force reflecting hand masters, are mechanical devices that provide kinesthetic feedback to human operators. The impression of haptic roughness can be effectively produced by applying force pulses to the hand, which can enrich user interactions in simulation or teleoperation. The nature of these pulses is determined by the physical characteristics of the contacting bodies, for example the "average" surface roughness of the pair. Various physical measures of surface roughness exist, and these can be used to produce the frequency and magnitude of the pulses necessary to depict the forces which would result from the actual sliding. However, the micro-modelling of actual collisions to create trajectories would be computationally prohibitive for real-time applications. It will suffice to produce haptic stimulation which gives the correct psychological illusion given the limitations on the user's perceptual acuity and ability to spatially and temporally connect surface features to individual haptic events. Thus, although the haptic device can not generally accurately track the roughness-induced micro-motion of the actual system trajectory, the interaction is enriched by depicting the essence of the scene events, with the limitations of the user's haptic input channels in mind. It is proposed that human visual acuity is unable to correlate the individual micro-surface details which cause the individual force impulses either spatially or temporally for typical surfaces. For example, consider some typical object sliding across a table where small textural impulse forces are haptically perceived, but due to the size, number and partial or complete visual obstruction of the actual surface asperities causing the force impulses, there is no spatial or temporal correlation between them by the individual. Therefore, the sense of texture can be treated as characterized by some macroscopic properties.

This paper will first present some relevant previous work in texture generation, both in haptics and in graphics. Then, a general discussion of surface and roughness properties is presented, followed by a discusion of the surface properties which cause haptic texture. A decomposition of contact forces is given followed by an algorithm to generate the haptic texture component of the contact force. Methods are then provided to allow the texture forces to be generated in low-level control layers, thus allowing them to run at high frequency. Finally, an implementation of the algorithm is presented.

---

*This work was supported in part by NSERC, the Institute for Robotics and Intelligent Systems (IRIS Project AMD-5), and the BC Advanced Systems Institute. The authors would like to thank Vincent Hayward for the Pantograph design, and Rod Barman, Stewart Kingdon, and IRIS project ISDE-6 for help in building the EMDC controller and software for the Pantograph.

2 Related Work

2.1 Textures in Haptics

Sakaguchi [Sak94] describes a system for automatically detecting and classifying surface texture, without duplicating the texture through a force display device. De Rossi [DR91] classified various sensory needs with respect to specific manipulation tasks, and described instrumentation technologies to measure surface properties in relatively unstructured teleoperation environments. Lederman et al. [LK87] studied the importance of hand motion in obtaining object information through haptic channels. Minsky et. al. [MOyS$^+$90] studied force display by creating texture maps consisting of spring approximations to local gradients of "bumps". They also investigated the correlation between visual and haptic acuity and some of the kinesthetic parameters relevant to force display, but their approach to texture force generation was deterministic.

2.2 Textures in Graphics

Texture generation and mapping has received considerable attention in graphics. As described by Heckbert [Hec86], the traditional graphical texturing problem comprises mapping a defined texture from some convenient space (called the texture-space), to the screen-space. Lewis [Lew89] surveys methods based on noise, while Perlin [Per85] [Per89] presents noise-based techniques which by-pass texture space. In Perlin's technique, a simple noise function is defined as

$$d = noise(x, y, z)$$

where the parameters x, y, z are object coordinates and d is applied at the time the object is rendered. By perturbing surface normals with the noise derivatives, various surface effects were presented, including "bumpy" and "stucco". As presented in this paper, the idea of stochastic texture generation also has application in haptic texturing.

3 Haptic Textures

3.1 Surface Roughness Model

Early work in the area of friction and surface roughness used a form of overlay model to represent surface texture. For example, Schurig [Sch40] defined a surface as consisting of "waviness", "roughness" and "surface flaws" (see Figure 1). Although Schurig described only roughness as being superimposed on waviness, the surface could be described as surface flaws overlain on roughness overlain on waviness. A statistical treatment is given in [Tho82]. The two prominent surface height characterization measures are root-mean-square deviation (R_q) and center-line average (R_a), which measure the variance of surface height asperities from a straight, average datum line.

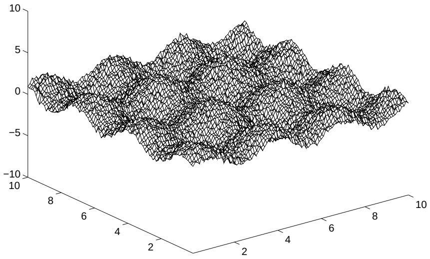

Figure 1: A "wavy" Surface Profile with Guassian Deviation Added

For our purposes, the most significant observation is that many surface preparation methods result in roughness profiles which are approximately Gaussian (see [Tho82]). Although practical surfaces can deviate from the Gaussian ideal, we will restrict our analysis to the Gaussian profile. The assumption is justified not merely because it adequately represents many surfaces, but because no attempt is being made to use the haptic device to generate a detailed, true trajectory of the motion of a sliding object; only a realistic representation of the surface texture is sought. Furthermore, many other details will be ignored, such as the contact patch size, which are necessary for a truely accurate analysis of surface interactions.

Consider a point moving laterally along the surface at speed v. If the surface height is sampled at points $v \times n \times dt$, where n is a positive integer and dt is the sampling period, a Gaussian histogram would result. Therefore, if our low-level control loop runs at time increments of dt, each control cycle should see a surface asperity with magnitude drawn from the Gaussian distribution. Note that this is independent of the speed at which the sampling occurs, since resampling a Gaussian distribution should result in a Gaussian distribution with the same mean and standard deviation. Therefore, the same Gaussian distribution is used regardless of how quickly the sliding pieces move across one another, and the mean and standard deviation are considered invariant contact properties.

3.2 Haptic Texture Definition

The forces of interaction between sliding surfaces are a complex function of the material properties and the surface geometry of the contacting bodies. The surface contact force can in general be decomposed into orthogonal normal and lateral components, and results from the contact and deformation of micro-surface asperities as two objects slide over each other under load. In modelling an object surface, decisions must be made about the level of detail in the model. For example, if using polygonal approximations, the relative size and number of polygons to include must be decided. The haptic texture is defined to be all the effects which are not explicitly accounted for by traditional rigid body contact normal (constraint) and lateral (friction) forces. To determine the level of detail, it is useful to consider the size of the contact patch between the sliding bodies. Texture forces will be caused by those surface profile components with spatial period lower than the dimensions of the contact patch. To facilitate the determination of appropriate texture generation parameters, we make the following assumptions:

- During sliding between surfaces, normal texture forces are generated due to interactions between surface asperities, with magnitude proportional to the height of the asperities.
- Lateral texture forces are generated proportional to the above normal forces.

The first assumption is justified when considering any two contacting surface asperitities sliding over each other. Force components are generated in normal and lateral directions as the asperities slide relative to each other. The second assumption follows from the first in that it is consistent with Coulomb friction. No attempt is made to model the individual surface asperity contacts which are produced during sliding. Such an approach is considered intractible not only because of the computational expense, but because conventional sensing and control technologies would have difficulty accurately reproducing any resulting trajectories to the small scales to which they would have to be computed. Instead, our approach will concentrate on isolating and using physically meaningful surface roughness parameters, with the result that the texture mapping will provide realistic feel to the user.

4 Force Decomposition

Figure 2 shows a decomposition of the contact force. The following model is used (see Appendix A for statistics notation):

$$\mathbf{F}_{contact} = \mathbf{F}_{constraint} + \mathbf{F}_{friction} + \mathbf{F}_{texture}$$

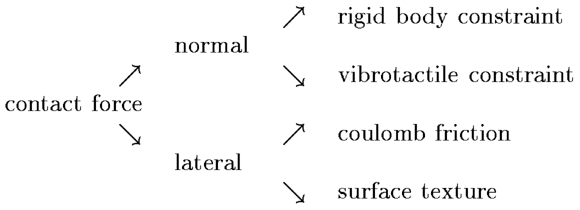

Figure 2: Force Decomposition

where

$$\begin{aligned} \mathbf{F}_{texture} &\sim u\ Normal({}^{F_{texture}}\mu, {}^{F_{texture}}\sigma^2) \\ u &= \begin{cases} 0 & \text{for } |v| < \epsilon \\ 1 & \text{otherwise} \end{cases} \end{aligned}$$

where v is the lateral speed, and ϵ is a small positive number. This provides a small deadband around a stationary position, improving stability. The choice of μ and σ will be determined by surface roughness characteristics and in the initial implementation, μ was taken as zero and σ was varied to produce different textural feels. It was predictably found that higher σ leads to a sensation of a rougher surfaces. The actual σ would ultimately be taken from surface roughness literature for different surfaces.

The normal constraint forces can be generated using a PD control approach (for instance, see Salcudean and Vlaar [SV94]). This component creates the illusion of surface hardness by producing forces which oppose interpenetration of simulated contacting bodies. With only this component, a frictionless, smooth surface, such as ice, is generated. The tangential Coulomb friction opposes tangential motion along the surface. Adding this component creates a smooth frictional surface, such as glass. The texture forces are then added to give surface roughness characteristics. They can be related to the R_q and R_a parameters in that both are statistical measures of variance, and the variance of the statistical haptic texture ($\mathbf{F}_{texture}$) is proportional to the R_q and R_a variances.

5 Generating Gaussian Texture

A realistic surface has many contacts occuring during motion. It is therefore desirable to compute the texture forces at relatively high frequencies. In our initial implementations a frequency of 1 KHz was used to produce effective textural feel.

The contacting bodies are generally described in a Cartesian frame. The statistical properties of the texture forces will thus be described in a Cartesian frame called the surface-space, with one axis perpendicular to the contact surface. These properties will then be projected to the world-space, which is a convenient inertial reference frame. From the world frame, the force properties are projected to joint-space. Once the texture force statistical properties are known in joint-space, actuation forces can be generated with the mapped statistical properties. Therefore, the necessary texture forces can be generated directly in joint-space, and there is no need to map individual forces from the surface space to the joint space. The reader is referred to the appendices for relevent statistical concepts.

5.1 Generating Texture Forces in World-Space

The texture properties are defined relative to an object's surface. The object's surface-space can thus be mapped to world-space. Let R be the orientation mapping from the surface-space to the world-space

$$B_w = RB_s$$

where B_w and B_s are the matrices whose columns are the unit vectors of the world-space and surface-space, respectively.

Then forces in surface-space map to forces in world-space according to

$$\mathbf{F}_w = R^T\mathbf{F}_s$$

where $\mathbf{F}_s$ is the force in surface-space, and $\mathbf{F}_w$ is the same force mapped to world-space.

Then, the statistical properties of $\mathbf{F}_w$ are as follows (see Appendix A).

$$\begin{aligned} {}^{F_w}\sigma^2 &= R^T({}^{F_s}\sigma^2)R && (1) \\ {}^{F_w}\mu &= R^T({}^{F_s}\mu) && (2) \end{aligned}$$

The properties of the texture can now be defined in the invariant surface-space and the stochastic behaviour mapped to the world-space. This removes the computational cost of mapping forces between spaces for each control cycle. If the lateral texture force variances are made functions of the normal constraint force, then, under the assumption that the change in normal force is low-bandwidth (since it is limited by the bandwidth of a human user), the computation of variance matrices is more efficient than producing and mapping individual forces - the normal force changes relatively slowly so the covariance matrix needs to be updated at low bandwidth.

5.2 Generating World-Space Forces in Joint Space

To implement the algorithm in low-level control, instead of generating the actual texture impulses in world-space one can generate only the necessary mean and covariance matrices for a given world-space region, and map those parameters to joint-space through the Jacobian. To achieve this, consider the following.

Let

$$\mathbf{\Gamma} = J^T\mathbf{F}$$

where $\mathbf{F}$ is a world-space force, and $\mathbf{\Gamma}$ is the corresponding joint-space torque vector. Restricting the analysis to three dimensional forces, and n dimensional joint-space,

$$\mathbf{\Gamma} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_n \end{bmatrix}, \mathbf{F} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

Then, if $\mathbf{F}$ has the form

$$\mathbf{F} \sim Normal({}^F\mu, {}^F\sigma)$$

$\mathbf{\Gamma}$ has the form

$$\mathbf{\Gamma} \sim Normal({}^\Gamma\mu, {}^\Gamma\sigma)$$

To obtain the joint-space covariance matrix, ${}^\Gamma\sigma^2$, begin by defining the covariance matrix for $\mathbf{F}$ as

$${}^F\sigma^2 = \begin{bmatrix} {}^f\sigma_{11} & {}^f\sigma_{12} & {}^f\sigma_{13} \\ {}^f\sigma_{21} & {}^f\sigma_{22} & {}^f\sigma_{23} \\ {}^f\sigma_{31} & {}^f\sigma_{32} & {}^f\sigma_{33} \end{bmatrix}$$

where ${}^f\sigma_{ij}$ is the covariance of the i'th and j'th components of the world force $\mathbf{F}$, as defined on the object's surface.

Then, the joint-space covariance matrix is given by

$${}^\Gamma\sigma^2 = \begin{bmatrix} {}^\tau\sigma_{11} & {}^\tau\sigma_{12} & \cdots & {}^\tau\sigma_{1n} \\ {}^\tau\sigma_{21} & {}^\tau\sigma_{22} & \cdots & {}^\tau\sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ {}^\tau\sigma_{n1} & {}^\tau\sigma_{n2} & \cdots & {}^\tau\sigma_{nn} \end{bmatrix} = J^T({}^F\sigma^2)J$$

To obtain ${}^\Gamma\mu$, define the vector of world-space force component averages

$${}^F\mu = \begin{bmatrix} {}^f\mu_1 \\ {}^f\mu_2 \\ {}^f\mu_3 \end{bmatrix}$$

Then,

$${}^\Gamma\mu = J^T({}^F\mu)$$

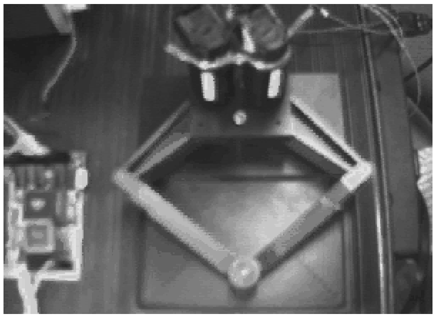

Figure 3: Pantograph Haptic Interface

Gaussian deviates can be generated in joint-space with the mapped mean and covariance and added directly to the actuation. This will be valid in so far as the computed Jacobian remains a good linear mapping between the world and joint spaces. In other words, the Jacobian should be recomputed only when it becomes a poor mapping.

5.3 Input Parameters to the Gaussian Texture Functions

The initial implementation used zero mean with different standard deviations, and the evaluation criterion was the subjective impression of a user. However, ultimately surface roughness parameters as quoted in surface roughness literature would be the appropriate values to use.

6 Implementation

6.1 Testbed

The algorithm was implemented on a two-degree-of-freedom Pantograph haptic interface designed by Hayward [RH94]. The device comprised two planar, two-link revolute-jointed arms hinged at the ends, with rotational actuators on each of the two base links, and a workspace of approximately 16cm wide × 10cm long (see Figure 3). Optical encoders provided actuator position feedback with 2000 counts/rev resolution, and actuation was achieved with pulse-width-modulated (PWM) amplifiers through 12 bit D/A conversion. A block diagram of the appratus is shown in Figure 4.

6.2 Results

Using this algorithm, it was possible to generate texture force pulses at approximately 1kHz using only a 16MHz fixed-point processor. Refering to Figure 4

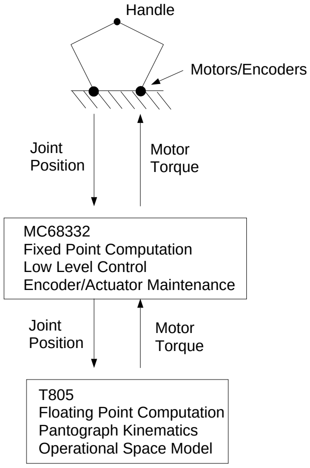

Figure 4: Implementation Apparatus

6.2.1 Lateral Texture Forces Only

A model with only lateral texture forces (similar to that of Minsky et al [MOyS$^+$90]) was implemented where the texture impulses were created with uncorrelated world-space components. Although a reasonable textural feel was produced to emulate very coarse texture, comparison with an object moving along an actual heavily textured surface (a piece of very coarse sand-paper) indicated that stick-slip phenomena form an important component for effective textural representation of very coarse surfaces. Representation of surfaces with less coarseness was found to be reasonably good.

6.2.2 A Straight Edge with Normal and Lateral Texture Forces

A straight edge was implemented combining normal constraint and texture impulses with tangential texture impulses. It was found that creating the texture impulse forces proportional to the normal constraint force created a more realistic feeling texture than making them independent. This is a reasonable result since higher deformation and micro-collision forces would be expected for larger normal constraint forces. The covariance between normal and lateral texture impulse components (non-zero off-diagonal terms in the world-space covariance matrix) did not produce a detectably different feel versus independent world-space texture impulses (diagonal world-space covariance matrix). This suggests that it is generally unnecesary to transform the texture from the object-space to world-space, and that a diagonal covariance matrix can be maintained in world-space. This saves the computation steps of mapping the texture from object-space to world-space, as well as making the joint-space mapping less costly since the world-space covariance matrix remains diagonal.

7 Conclusions

The paper describes a method for generating a realistic texture for haptic devices. The method is simple and can be extended to incorporate actual physical measures of surface properties to produce the appropriate textural feel. This opens the possibility of using online roughness sensing in teleoperation applications, as well as specifying physical parameters in simulation, to obtain the appropriate textural feel in the haptic device. Initial implementations have shown that the method can produce realistic haptic texture.

A Relevant Elementary Statistical Concepts

Consider a system defined by

$$\mathbf{Y} = A\mathbf{X}, \mathbf{X} \in \mathbb{R}^m, \mathbf{Y} \in \mathbb{R}^n, A \in \mathbb{R}^{n\times m}$$

If the components of $\mathbf{X}$ are samples from a statistical distribution, then the system has the following statistical properties.

A.1 Notation

A scalar variable with Gaussian or normal distribution is denoted $x \sim Normal(\mu, \sigma)$, where the mean and variance of the population from which x is taken are μ and σ, respectively. For a vector $\mathbf{X}$, each of its components can be taken from a different population, with notation $\mathbf{X} \sim Normal({}^X\mu, {}^X\sigma^2)$.

The mean and covariance of $\mathbf{X}$ are respectively denoted by

$${}^X\mu = \begin{bmatrix} E[x_1] \\ E[x_2] \\ \vdots \\ E[x_m] \end{bmatrix}, \quad {}^X\sigma^2 = \begin{bmatrix} {}^x\sigma_{11} & {}^x\sigma_{12} & \cdots & {}^x\sigma_{1m} \\ {}^x\sigma_{21} & {}^x\sigma_{22} & \cdots & {}^x\sigma_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ {}^x\sigma_{m1} & {}^x\sigma_{m2} & \cdots & {}^x\sigma_{mm} \end{bmatrix}$$

where E is the statistical expectation, and ${}^x\sigma_{ij}$ is the covariance between the i'th and j'th components. Upper-case leading super-scripts indicate a vector and refer to vector statistical properties, while lower-case leading super-scripts are scalars and refer to the statistical properties of individual vector components.

A.2 Covariance

If the covariance matrix for $\mathbf{X}$ is ${}^X\sigma^2$ then the covariance matrix for $\mathbf{Y}$ is given by ${}^Y\sigma^2 = A({}^X\sigma^2)A^T$.

A.3 Expectation (Mean)

If the mean of $\mathbf{X}$ is given as ${}^X\mu$ then the mean of $\mathbf{Y}$ is given by ${}^Y\mu = A({}^X\mu)$.

A.4 Generating Random Deviates with Given Statistical Properties

Numerous algorithms exist for generating Gaussian deviates, and many are presented in [PTVF92] and [Knu81], to which the reader is referred.

References

[DR91] D. De Rossi. Artificial tactile sensing and haptic perception. *Measurement Science & Technology*, 2(11):1003–16, 1991.

[Hec86] P.S. Heckbert. Survey of texture mapping. *Computer Graphics and Applications*, 6(11), 1986.

[Knu81] Donald E. Knuth. *The Art of Computer Programming*, volume 2. Addison-Wesley Publishing Company, 2 edition, 1981.

[Lew89] J.P. Lewis. Algorithms for solid noise synthesis. *Computer Graphics*, 23(3), July 1989.

[LK87] S.J. Lederman and R.L Klatzky. Hand movements: A window into haptic object recognition. *Cognitive Psychology*, 19(3):342–68, 1987.

[MOyS+90] M. Minsky, M. Ouh-young, O. Steele, F.P. Brooks, Jr., and M. Behensky. Feeling and seeing: Issues in force display. *Computer Graphics*, 24(2):235–43, March 1990.

[Per85] K. Perlin. An image synthesizer. *SIGGRAPH'85*, 19(3), July 1985.

[Per89] K. Perlin. Hypertexture. *Computer Graphics*, 23(3), July 1989.

[PTVF92] W.H Press, S. A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 2 edition, 1992.

[RH94] C. Ramstein and V. Hayward. The pantograph: A large workspace haptic device for a multi-modal human-computer interaction. *Conference on Human Factors in Computing Systems ACM/SIGCHI*, 1994.

[Sak94] Y. Sakaguchi. Haptic sensing system with active perception. *Advanced Robotics*, 8(3):263–283, 1994.

[Sch40] O.R. Schurig. How should engineers describe a surface. *Proceedings of the Special Summer Conferences on Friction and Surface Finish, MIT*, June 1940.

[SV94] S.E. Salcudean and T.D. Vlaar. On the emulation of stiff walls and static friction with a magnetically levitated input/output device. Technical report, Dept. of Electrical Engineering, University of British Columbia, Vancouver, BC, Canada, March 1994.

[Tho82] T.R.(Editor) Thomas. *Rough Surfaces.* Longman Group Ltd., 1982. London.