Perceptually Augmented Simulator Design Through Decomposition

Timothy Edmunds[*]
Rutgers University

Dinesh K. Pai[†]
University of British Columbia

Abstract

We approach the problem of determining a general method for augmenting haptic simulators to amplify the perceptually salient aspects of the interaction that induce effective skill transfer. Using such a method, we seek to simplify the design of haptic simulators that can improve training effectiveness without requiring expensive improvements in the capability of the rendering hardware.

We present a decomposition approach to the automated design of perceptually augmented simulations, and we describe a user-study of the training effectiveness of a search-task simulator designed using our approach vs. an un-augmented simulator. The results indicate that our decomposition approach allows existing psychophysical findings to be leveraged in the design of haptic simulators that effectively impart skill by targeting perceptually significant aspects of the interaction.

1 Introduction

In previous work [1] we established that augmenting perceptually salient interaction features can improve haptic simulations without requiring improved rendering hardware. We used this approach to design an augmented simulator for a surgical task (bone-pin insertion), and showed a significant task improvement among subjects who trained on that simulator. However, we left open the question of how this design process could be generalized or automated.

The wide variety of haptic tasks and the multiple modes of haptic interaction make it hard to address this general problem. In this paper, we propose an approach that makes the general augmented-design problem more tractable through decomposition.

As a user performs a task, various different interaction features are encountered. For example, when a mechanic inserts an engine part into a visually obscured location, the interaction will involve the shape and surface properties of the part and of the engine, as well as transient features such as making/breaking contact between the part and the engine, stick/slip as the part slides into place, or jamming if the part is inserted incorrectly. But the features of the interaction that are pertinent to the user depend on what aspect of the task is being performed. For the mechanic, contact between the part and the engine may be irrelevant when maneuvering the part towards the general area of insertion, but is critical to correctly insert the part precisely in place. The high-level task can be decomposed into subtasks that correspond to different contexts for interaction. In our example, these subtasks might be: manipulating the part to acquire a secure grasp and assess its shape; maneuvering into the general area; exploring the area of insertion to find the correct insertion point; positioning the part for insertion; and sliding the part into place. The subtask being performed determines which interaction features are most perceptually pertinent and need to be effectively rendered by the simulation (e.g., when the user is sliding the part into place, effective rendering of stick/slip and jamming is critical).

If we can identify the subtask being performed by the subject at any given time, then we can selectively augment the interaction features that are deemed pertinent for that subtask. We have thus reduced the problem of holistically assessing an interaction in progress and generating appropriate augmentation to three subproblems: decomposing the overall task into subtasks; determining what augmentation is appropriate for the perceptual context of each subtask; and detecting throughout the interaction what type of subtask is being performed.

A challenge in investigating the broad problem of training simulator design is that experimentation requires a laboratory task that captures aspects of real-world tasks while being repeatable and allowing detailed analysis of the interaction. Towards this end, we designed an artificial haptic search task that mimics the activities found in an engine-part insertion task: the subject has to scan the haptic environment to find textured surface patches; identify the surface patch with the correct texture; and precisely locate its centre. We use this task as the basis for an evaluation of our design approach.

Our Contributions: In the work described here, we apply our decomposition approach to the problem of general augmented training simulator design. We conducted a user-study to validate our approach; we created a haptic search task that involves a variety of interactions, and used our decomposition framework to design an augmented simulator for training on the task. We analyzed the training effectiveness of the augmented simulator across a variety of metrics designed to gauge the effective skill transfer of different components of the augmented simulation. Our results indicate that the augmented simulation design that was generated by our generalized decomposition approach resulted in an effective training simulator.

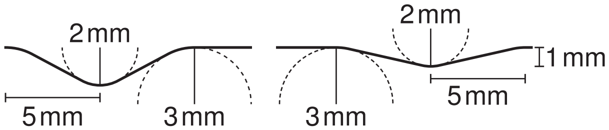
Background and Approach

One approach to the problem of automatically generating augmentation for haptic simulations is to measure the haptic properties of the real task, find the differences between those properties and the properties rendered by the simulation, and augment the simulation with the "difference" between the two. Acceleration matching for impact augmentation [5] is an example of this type of approach. However, while this approach might help achieve greater fidelity, that criterion is not always the best one for judging the effectiveness of a haptic simulation; when the goal of the simulation is to improve transfer of training, controlled *deviation* from the real dynamics can improve the simulation's effectiveness [14]. Instead of a purely fidelity-based evaluation criterion, we need to consider what augmentations will achieve the desired training effect in a haptic simulation. For example, to evaluate the effectiveness of a surgical trainer, the procedure success rate after training is more important than the amount of error in the forces rendered.

The problem of augmenting haptic simulations is complicated by the large number of different dimensions that are perceived haptically. Early work on haptic exploration (focussing primarily on haptic identification) [4] identified a set of haptic dimensions that are directly sensed and aid in the haptic identification of objects. (A haptic dimension [7] is a domain of variation that is accessible to the perceptual system — e.g., hardness of a surface being tapped.) The identified dimensions were texture, hardness, temperature, weight, global volume, exact shape, part motion, and specific function. The researchers also identified a set of *exploratory procedures* that are typically used to assess an object's value along each of these dimensions.

[*] e-mail:tedmunds@cs.rutgers.edu

[†] e-mail:pai@cs.ubc.ca



(a) Groove Cross-Section (b) Pit Cross-Section

Figure 1: Environment geometry at 2.5× scale. (a) The cross-section of the groove surrounding each scene element. (b) The cross-section of the pit at the centre of each scene element.

In the broader context of general tasks, we can define the set of *interactive procedures* as a super-set of these exploratory procedures. Where the goals of all exploratory procedures are to investigate and assess a haptic dimension, not all recognizable actions fall into this category. A review of the haptic literature reveals many examples of identifiable interactive procedures. Some are procedures intended to assess properties of the environment, such as feeling a spot with a finger to determine the small-scale geometry [11], tapping or scraping an object with a tool to estimate its material properties [5, 9], or moving an object's parts to determine its range of motion [4]. Other procedures involve spatial localization, such as multi-finger touching to detect where a specified feature occurs [10], sliding a finger along a line to find where a haptic target lies amidst distractors [8], or manually scanning visually obscured objects to locate a goal [6]. There are also procedures whose intent is to modify the environment to accomplish a goal, such as guiding a catheter into a vessel [2], or inserting a peg into a hole [13].

A given task may involve the exercise of any combination of interactive procedures, either simultaneously or sequentially, in separate subtasks. By focussing on the set of interactive procedures used in performing a task, we can guide the automatic generation of augmentation to improve the performance of the task.

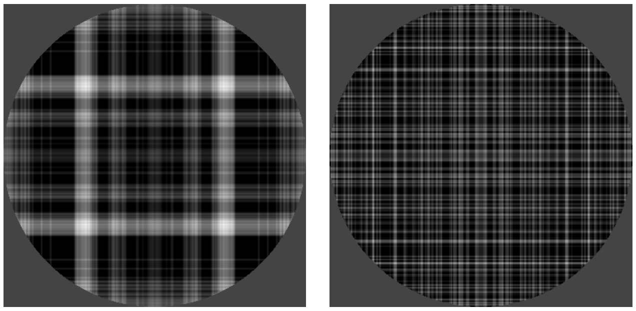
As well as determining which subtasks are performed in executing a task, we need to consider what *augmentations* are appropriate to improve the skill transfer for the subtasks. Here we can again leverage findings in the literature that illuminate how different interactive procedures are affected by properties of the environment.

The remainder of this paper is organized as follows. In Section 2 we describe the haptic search task that we developed to facilitate a concrete investigation of our approach to simulator design. We give the details of how we applied our decomposition-based approach to generate an augmented training simulator for the haptic search task in Section 3. In Section 4, we describe the user study we conducted to evaluate the training effectiveness of our augmented simulator. In Section 5, we draw conclusions about the effectiveness of our proposed decomposition-based approach, including guidelines for the design of perceptually-augmented haptic training simulators.

2 Haptic Search Task

To allow for a concrete investigation of the design of virtual simulators for real tasks, we need to test our approach on a specific task. We created a haptic search task that is structurally similar to the mechanic's problem of inserting an engine part without visual feedback. The task parallels the standard visual search task commonly used in psychophysical experiments: the subject attempts to locate a target stimulus that is presented in the company of distractor stimuli that have similar (but distinguishable) characteristics. In our search task, the subject must: haptically scan the environment to search for a target (or distractor) texture patch; discriminate between the target and distractors based on texture properties; and finally locate the precise centre of the target patch.

This search task is useful in the laboratory setting because it captures key aspects of real-world tasks (such as including a sequence of actions that must be performed to allow later phases of the task to be completed) while allowing detailed recording of the subject's interaction to support rigorous analysis of the task performance along multiple dimensions. This task is also easily repeatable for user studies because it can be implemented entirely in a virtual environment.



(a) Distractor (b) Target

Figure 2: Example (a) distractor and (b) target texture patches at 1:1 scale. The texture is represented visually by lightness corresponding to the coefficient of friction, μ (ranging from 0 to 0.75).

2.1 Stimulus Design

In order to apply and evaluate our decomposition approach to developing an augmented training simulator for the haptic search task, we created a virtual environment implementation of the search stimulus.

The environment for the haptic search task consists of a 3-D workspace with smooth flat walls around four sides of a 240 mm by 240 mm floor whose height and surface roughness are varied to create the target and distractor stimuli.

The floor is a height field that is uniformly zero everywhere outside a target or distractor (a *scene element*). Each scene element consists of a groove surrounding a flat circular patch of roughly textured surface. At the centre of the patch, there is a small pit. See Figure 1 for a visualization of the cross-sectional geometry of the scene elements. The textured patch has a radius of 20 mm, and the surrounding groove is 10 mm wide. The groove's cross-section is smooth, with its bottom being a segment of a circle, and each lip being segments of circles. The pit at the centre is similarly smooth, with a total radius of 5 mm and a maximum depth of 1 mm at the centre. The only difference between targets and distractors is the texture of the patch (manifested as variation in the coefficient of friction, which is functionally equivalent to texture, but more robust [9]). We use the standard Coulomb friction model, $\mathbf{f}_f = -\mu\|\mathbf{f}_n\|\mathbf{u}_m$, where $\mathbf{f}_f$ is the frictional force, $\mathbf{u}_m$ is a unit vector in the direction of motion, and μ is the (spatially varying) coefficient of friction. Both types of texture are generated by adding a baseline coefficient of friction (μ_0) to the output of a noise-driven autoregressive AR(p) process that generates the randomness and periodicity typical of real surfaces [9]:

$$\mu(x,y) = \mu_0 + \tilde{\mu}(x,y) \tag{1}$$

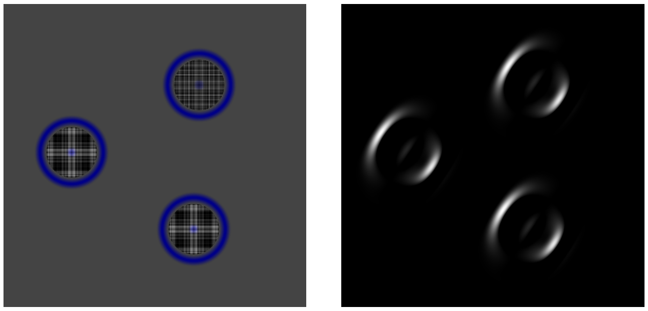
$$\tilde{\mu}(x,y) = \tilde{\mu}(x\Delta_x) + \tilde{\mu}(y\Delta_y) \tag{2}$$

$$\tilde{\mu}(k) = \sum_{i=1}^{p} a_i \tilde{\mu}(k-i) + \sigma\varepsilon(k) \tag{3}$$

where Δ_x and Δ_y are the spatial discretization resolutions in the x and y direction, k is the sample index along a dimension, σ is the standard deviation of the input noise, and $\varepsilon(k)$ is a zero mean noise input with a standard deviation of one. For both the target and distractor textures, we used an AR(2) model, and a spatial discretization resolution of 10 samples/mm. See Table 1 for the parameters for the AR(2) functions used to generate the target and

Target		Distractor	
μ_0	0.1	μ_0	0.1
a_1	0.783	a_1	0.25
a_2	0.116	a_2	0.1
σ	0.05	σ	0.1

Table 1: Texture parameters.



(a) Example stimulus

(b) Processed Height Field

Figure 3: (a) A visual rendering (at 1:6 scale) of a stimulus environment presented to the user. The texture is represented visually by lightness corresponding to the coefficient of friction, μ, and the height map of the surface is overlayed in blue. (b) The result of convolving the example height field from (a) with the Gabor filter shown in Figure 4. This image is one slice of a 32×5×1024×1024 lookup table for the scan augmentation.

distractor textures. Examples of the texture patches generated by these parameters are shown in Figure 2.

The surface outside the texture patches has a uniform coefficient of friction $\mu_{background} = 0.2$.

The full stimulus for one episode of the search task consists of two distractors and one target (see Figure 3a). The scene elements are equally spaced around a circle of radius 66.7 mm; the only variation between episodes is the orientation of the triangle described by the three scene elements (and the noise driving the autoregressive texture of the elements).

2.2 Stimulus Interaction

To realize a repeatable haptic search task, we implemented a virtual environment that allows the subject to interact with the stimulus through a SensAble Technologies PHANTOM Premium 1.0 haptic device with 6 degrees of freedom in position input and 3 degrees of freedom in force output.

Interaction with the environment is simulated by a quasi-static system where the stylus tip of the PHANTOM represents the master position that is spring-coupled to a proxy point that is constrained to lie within the workspace and above the surface of the floor. When the master is inside the walls and above the floor, the proxy moves with the master and no forces are generated.When the master is outside the walls or below the floor, the proxy is placed at the closest permitted point, and a spring force acts on the master to pull it toward the proxy: $\mathbf{f}_n = k(x_p - x_m)$, where k is the stiffness of the virtual spring, and $\mathbf{f}_n$ is the normal force applied to push the master toward the surface.

The texture of the scene elements (and the friction of the background) is implemented by a stick-slip Coulomb model as described by Salisbury et al. [12]. In that model, the tangential force is affected by the local coefficient of friction and the normal force; hence the frictional force is also affected by the spring stiffness k.

Each episode of the search task begins with the master held in place (by a stiff spring force) in the centre of the workspace, 20 mm above the surface. Once the episode begins the spring force is released, and the subject is free to explore the environment. The goal of the task is for the subject to locate the centre of the target scene element and hold the stylus tip there for 0.5 seconds.

2.2.1 Visual Stimulus

In addition to the haptic feedback described above, the subject is presented with a few visual cues: the outline of the boundaries of the workspace, and the position of the proxy. This sparse visual rendering of the environment is displayed on a vertical screen in front of the subject. This allows the subject to use visual cues to construct a spatial representation of the location of haptic features (which are not displayed) as they are felt.

3 Augmentation

Having defined the stimulus for our haptic search task, we created a basic training simulator that is simply the same rendering algorithm as the real task but with artificially degraded stiffness (corresponding to the general design condition in which the rendering hardware cannot trivially reproduce a real interaction with high fidelity). We then applied our approach to develop an augmentation scheme for this simulator. In our approach to automatic simulation augmentation, a complete task is decomposed into subtasks for which different augmentation is applied in accordance with the perceptual features involved in executing the subtask.

3.1 Task Decomposition

As discussed in Section 1, the decomposition can be assisted by focussing on interactive procedures. For our haptic search task, we were able to use this assistance to create a subtask decomposition with limited domain knowledge.

The first subtask that the subject must execute is to locate a scene element; we call this the *scan* subtask. In this subtask, the subject typically scans the surface with large scale, high-speed motions, until he or she detects the high-temporal-frequency force discontinuity event that signals that the stylus has encountered (the rising slope of) a groove around a scene element.

The second subtask is assessing the shape (and thus the extent of the texture patch) of the scene element. In the *shape assessment* subtask, the subject traces part or all of the groove around the scene element to generate a spatial representation of where the texture patch (and its centre) lies.

The other subtask is the *identification* subtask; having located a scene element, the subject must explore it (with a scrubbing exploratory procedure) to gauge the roughness of the surface in order to identify the scene element as a distractor or target.

Although there there are only three subtasks in this decomposition, a single execution of the overall task can include multiple instances of each subtask in different orders. The subject may scan, identify a distractor, scan again, assess shape, identify the target, and reassess the shape before moving to the target's centre.

3.2 Subtask Augmentation

Having identified the different subtasks that make up our haptic search task, we need to assign augmentations for each subtask.

3.2.1 Scan Augmentation

In the **scan** subtask, the pertinent perceptual features of the interaction are the force discontinuities experienced when the stylus tip passes over areas of changing height. Since the simulator has low stiffness haptic feedback, these high-temporal-frequency force discontinuities are lost. These perceptual features can be restored through the use of open-loop augmentation generated by automated techniques similar to those used in computer vision.

A common technique for processing images to extract or highlight pertinent features is to convolve the image with a filter. Since we want to identify places in the environment where force discontinuities are experienced during scanning, our problem is similar to that of edge detection. Rather than edge detection in the 2-dimensional (x,y) space though, we are performing edge detection in the 4-dimensional (x,y,v_x,v_y) space of the interaction between the height of the surface at (x,y) and the velocity of the stylus.

We can think of the height map of the surface as an image whose edges we want to find, where for a particular stylus velocity we are only interested in edges of a certain orientation and spatial frequency (i.e., at higher speeds, we want to detect edges with lower spatial frequency). The 2-dimensional anti-symmetric Gabor filter is a widely used convolution kernel for oriented edge detection at configurable spatial resolution:

$$g(x,y,\theta,\sigma) = \exp\left(-\frac{x'^2+0.25y'^2}{2\sigma^2}\right)\cos\left(\pi\frac{x'}{2\sigma}+\frac{\pi}{2}\right) \quad (4)$$

$$x' = x\cos\theta + y\sin\theta \quad (5)$$

$$y' = -x\sin\theta + y\cos\theta \quad (6)$$

Here θ is the orientation of the filter (direction perpendicular to the parallel stripes), and σ is the standard deviation of the Gaussian envelope that determines the spatial resolution of the filter. See Figure 4 for an example of the type of filter used.

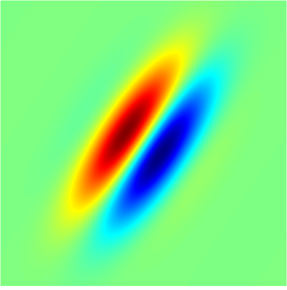


Figure 4: Example Gabor filter ($\sigma = 2^4$ px, $\theta = 0.589$ rad, $\gamma = 0.5$).

By pre-computing the convolution results of the surface's height map with Gabor filters of various orientations and spatial resolutions, we can create a 4-dimensional lookup table that indicates which surface locations (at a given stylus velocity) should trigger a haptic pulse to signal an edge-crossing.

For the augmented training simulator, we pre-computed the convolution of each stimulus with Gabor filters at 32 different (equally spaced) orientations and 5 different scales ($\sigma = 2^0, 2^1, \ldots, 2^4$, where σ is in units of pixels, and the height map of the stimulus is represented as a 1024x1024 image — see Figure 3b). During the scan subtask, the stylus tip location and velocity are used as indices into a lookup table formed by all 160 pre-processed images for the current stimulus; if the lookup value exceeds a threshold, an open-loop fixed-width force pulse is initiated (upwards).

3.2.2 Shape Assessment Augmentation

In the **shape assessment** subtask, the subject follows the groove around a scene element to determine the spatial extent of the element (and the location of its centre). This is an example of an exploratory procedure that uses the environment to constrain and guide the exploratory motion. Since this exploratory procedure leverages the curvature of the surface (which produces the constraints on motion), we augment the simulation for this subtask by applying local force-fields based on surface curvature.

The motion constraints imposed by curved surfaces channel motion towards points (or paths) that are local minima of surface curvature (i.e., points of maximum concavity). By constructing force fields that attract the proxy toward these loci of minimal curvature, the guidance used by the shape-exploration procedure can be replicated in the low-stiffness simulator.

In order to simplify matters computationally (and to match the local effect of curvature-induced constraints), we want force-fields that have bounded extent and that are smooth, so as to avoid instability. We choose a single-cycle cosine function:

$$\mathbf{f}_{shape} = \begin{cases} f_{max}\left(1-\cos(2\pi\frac{d}{d_{max}})\right)\frac{1}{r_{curv}}\hat{\mathbf{n}} & \text{if } d \le d_{max} \\ 0 & \text{if } d > d_{max} \end{cases} \quad (7)$$

where f_{max} is a parameter controlling the overall scale of the augmentation force (we used f_{max} = 0.75 N), d is the distance to the nearest local minimum of curvature, d_{max} is the distance threshold imposed to make the force-fields local in extent (we used d_{max} = 5 mm), $\hat{\mathbf{n}}$ is a unit vector towards the attracting point, and r_{curv} is the radius of curvature (along $\hat{\mathbf{n}}$).

3.2.3 Identification Augmentation

In the **identification** subtask, the subject uses the lateral motion exploratory procedure to assess the roughness of the surface. Klatzky and Lederman [3] found that when perceiving roughness through a probe (as when perceiving roughness from direct skin contact), humans are able to achieve some measure of speed constancy in their perception of the vibratory phenomena induced by surface roughness (i.e., roughness is judged not by vibratory frequency alone, but by speed-normalized vibratory frequency).

Since the subject's perception of the surface roughness is affected by the speed of the subject-controlled motion, it is insufficient to simply augment the identification subtask by applying open-loop vibration at a fixed frequency. Instead, we wish to produce vibratory effects that mimic those of high-stiffness texture interaction, independent of speed. To achieve this, we can work in the speed-independent space of the original texture.

In the friction coefficient variation model of texture, the vibration experienced during lateral motion over texture is due to changes in the coefficient of friction; therefore, we augment the identification task by applying vertical forces proportional to the change in the coefficient of friction.

When the proxy is in contact with the surface, we look up the coefficient of friction, but instead of using it to generate tangential forces (which, in the low stiffness simulator fail to convey the surface texture), we compare it to the previous coefficient (i.e., the coefficient at the previous time-step of the rendering cycle), and generate a vertical force proportional to the change in coefficient (independent of the normal force). Although this generates vertical forces, rather than lateral friction forces, the vibratory signal experienced through the stylus conveys the same frequencies.

3.3 Subtask Identification

Having decomposed our task into subtasks and selected the augmentation for each subtask, the remaining problem is to identify during the interaction which subtask is being performed, and which augmentation(s) should thus be active.

Here we once again leverage the coupling between subtask and interactive procedure; since different procedures are used to accomplish different subtasks, we can identify the subtask that the subject is attempting to perform by identifying the interactive procedure being used. In our case, we can distinguish between the scanning, tracing, and scrubbing procedures used respectively in the scan, shape assessment, and identification subtasks, based solely on position in the environment and velocity thresholds.

Since scanning is a relatively high-speed motion, the scan augmentation is only activated if the stylus speed is at least 468.75 mm/s (the speed corresponding to the smallest size of Gabor filter used to preprocess the height map).

Since the shape assessment procedure is executed using finer-controlled (slower) actions than the scan procedure, the shape assessment augmentation is activated when the stylus speed is below the scanning augmentation threshold. Of course, since the force fields are local in extent and are located around the curvature minima, the shape assessment augmentation is only truly active when the subject is exploring pertinent surface geometry.

The identification subtask is characterized by the use of lateral motion to investigate an area of surface texture. The identification augmentation is thus only activated when the stylus tip is in contact with a textured surface, and only when the stylus is "moving laterally." In the context of a discrete time-step rendering loop this lateral motion criterion is deemed to be satisfied when the stylus's tangential speed is at least enough to move it from one texel (at the spatial discretization Δ) to the next (i.e., $\mathbf{v}_{x,y} \ge \sqrt{2}\Delta$).

4 User Study

To test the effectiveness of the augmented simulator design generated by our approach, we conducted a user study comparing the augmented simulator against the basic unaugmented simulator. Twelve subjects (recruited from faculty and students in the Rutgers Computer Science and Psychology departments) were included in the experiment. All subjects gave written consent and were compensated for their time. All subjects were right-handed and used the PHANTOM with their right hands.

A subject's participation consisted of three blocks of trials taking place in two sessions on different days (see Table 2). In the first session, each subject was familiarized with the capabilities of the PHANTOM device and the task to be performed; the subject then performed a baseline block of 50 episodes of the search task.

Block	Group 1	Group 2
Baseline	Full Stiffness	Full Stiffness
Training	Low Stiffness	Augmented Low Stiffness
Evaluation	Full Stiffness	Full Stiffness

Table 2: The simulations used by the subjects in each of the blocks of the experiment.

In the second session, each subject performed 50 training episodes on one of two simulations of the search task (with subjects randomly assigned to a simulation group), followed by 50 episodes of evaluation on the "real" search task.

The first simulation was simply a degraded version of the rendering of the real search task (the stiffness coefficient k was set to $60.0\ \frac{\mathrm{N}}{\mathrm{m}}$ in contrast to the rendering of the real task that used the device's nominal max stiffness of $600\ \frac{\mathrm{N}}{\mathrm{m}}$). The second simulation was an augmented version of the degraded simulation. The low stiffness was still used for the quasi-static force output, but active augmentation was applied according to the subtask being performed.

4.1 Results and Analysis

Since all blocks of trials in this experiment were performed on a simulated virtual environment, complete captures of the stylus position and forces were recorded for all trials, allowing multiple dimensions of the subjects' performances to be analyzed. As well as comparing overall task improvement after training, we measured and compared indicators of proficiency for each of the identified subtasks. While the success rate on the task doesn't significantly differ between the augmented and unaugmented training groups, analysis of the subtask proficiency metrics indicates that the effort required to achieve post-training success is significantly larger for the group trained on the unaugmented simulator.

4.1.1 Task Success

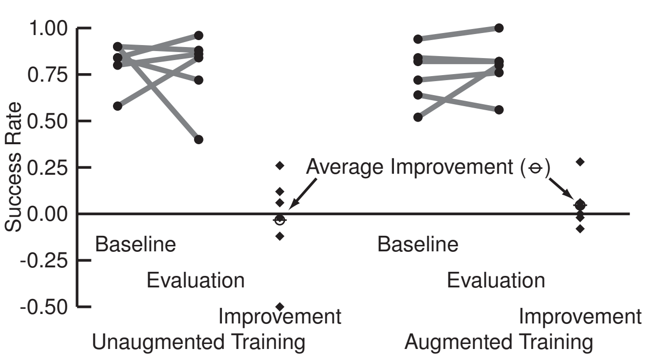


Figure 5: Baseline success, evaluation success, and change in success rate are plotted for each subject. On the left is the group of subjects that trained on the unaugmented simulation. On the right is the group of subjects that trained on the augmented simulation.

A primary metric for evaluating the effectiveness of the proposed augmentation technique is a comparison of the improvement in rate of successful task completion after training on the augmented vs. the unaugmented training simulation. For each subject, we measured separately the rate of successful task execution (i.e., finishing the trial by selecting the target scene element rather than one of the distractors) before and after simulator training:

$$\text{success rate} = \frac{\text{successful executions}}{\text{total executions}} \tag{8}$$

$$0 \leq \text{success rate} \leq 1 \tag{9}$$

We compared the success rate before and after simulator training to determine the subject's absolute improvement.

$$\text{improvement} = \text{success rate}_{after} - \text{success rate}_{before} \tag{10}$$

$$-1 \leq \text{improvement} \leq 1 \tag{11}$$

Figure 5 shows the success rates and improvement of each subject, grouped by whether the subject was trained on the augmented or unaugmented training simulation.

The group that trained on the unaugmented simulation had an average improvement of -0.03 ($\sigma = 0.26$), and the group that trained on the augmented simulation improved by an average of 0.05 ($\sigma = 0.12$). While this indicates that the subjects who trained on the unaugmented simulation improved more than the other subjects, a two-sample Kolmogorov-Smirnov (K-S) test (with the null hypothesis that unaugmented group's improvement cumulative distribution function is larger than that of the augmented group) yields an asymptotic p-value of 0.4425, so these results alone are not necessarily statistically significant (as indicated by the relatively large standard deviations of the samples).

4.1.2 Scan Subtask

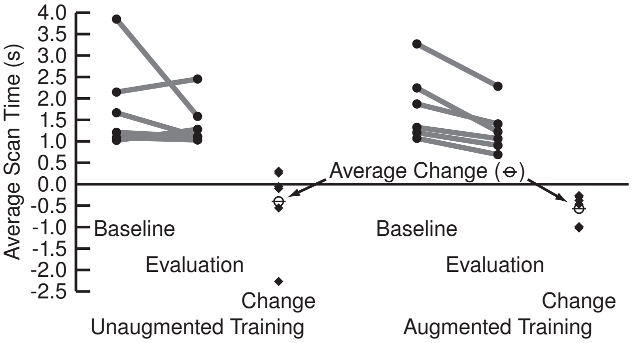


Figure 6: Change in average time to first encounter with a scene element before and after training.

To evaluate the isolated effectiveness of the scan subtask augmentation, we looked at the time between the start of the trial and the subject's first encounter with one of the scene elements, which we take to be indicative of both the subject's proficiency at the scan subtask and the subject's confidence in his or her sensitivity to the haptic stimulus (since faster exploratory movement indicates that the subject expects to be able to detect higher-frequency changes in the haptic stimulus).

For each subject, we measured the average time between the start of a trial and the first encounter with a scene element for both the baseline block of trials and the evaluation block. We then computed the absolute change in that average from before to after training. The change for each subject (sorted by training type) is shown in Figure 6. These results show a statistically significant training effect (the K-S asymptotic p-value is 0.0383).

4.1.3 Identification Subtask

Although a subject's proficiency at the identification subtask is strongly indicated by the overall task success rate, we can also gauge the subject's self-assessment of his or her skill at the identification subtask by examining the number of scene elements that the subject explores.

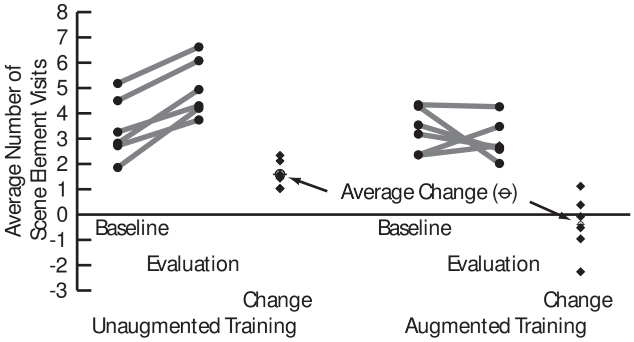


Figure 7: Change in average number of scene elements explored after first exploring the target element.

We computed the average number of scene element explorations that a subject performed after first visiting the target scene element in the baseline trial block and the evaluation block. The absolute change in average number of scene element visits is plotted in Figure 7. These results show a statistically significant training effect (the K-S asymptotic p-value is 0.0061).

4.1.4 Shape Assessment Subtask

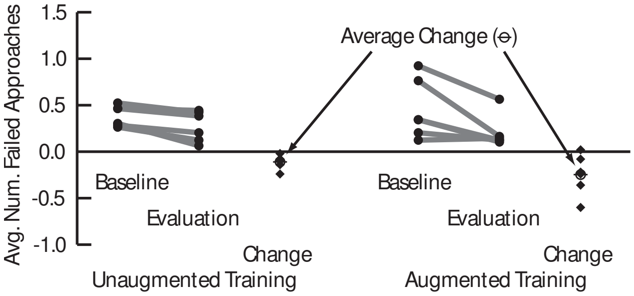


Figure 8: Change in average number of failed attempts to locate the centre of the scene element.

Since the shape assessment subtask is used to provide the subject with a spatial representation of the location of the centre of the scene element, a good indirect indicator of the subject's skill at performing this subtask is the number of attempts needed to locate the centre after the subject has decided to select a scene element.

We computed each subject's average number of failed approaches preceding the final successful approach for the baseline trial block and the evaluation trial block. The absolute change in average number of failed approaches is plotted in Figure 8. This metric shows a slight training effect, but is not statistically significant (the K-S asymptotic p-value is 0.1597)

5 Conclusions

In the study described in this paper, only one dimension of task performance (correct discrimination between target and distractor patches) was made explicit to the subjects, but we were also able to analyze subject performance (and improvement) along other dimensions. This analysis indicated that the augmentations that were incorporated on the basis of existing psychophysical findings were effective at improving the ability of subjects to locate the scene elements, at speeding the discrimination decision (in terms of number of redundant visits to scene elements), and at developing more control in the subjects' approach to the scene element centre once the discrimination choice was made.

The study results also raise some questions for future work. One interesting result is the apparent effect of the degraded training on subject behaviour indicated by the identification subtask results; the subjects who trained on the unaugmented simulator all became more cautious (in terms of the amount of exploration before making a decision). Here the *paucity* of the simulation seems to have had a strong training effect (though the increased caution did not translate into increased average success rate).

We can extract some specific guidelines for interactive simulation design from our results. The simulated task can be decomposed into subtasks based on domain knowledge and guided by analyzing the task in terms of the interactive procedures used in performing the task. Augmentation can then be designed for each subtask by reinforcing the haptic dimensions applicable to the relevant interactive procedures (this reinforcement does not have to create higher fidelity with the real experience). At run-time, the interactive procedure being performed can often be detected by simple methods (e.g., even temporally complex procedures like back-and-forth scraping for roughness examination and scanning to locate haptic features can be distinguished by basic velocity thresholds).

Acknowledgements: This work is funded in part by a Bevier Fellowship, NSF, Canada Research Chairs Program, and NSERC.

References

[1] T. Edmunds and D. K. Pai. Perceptual rendering for learning haptic skills. In *Haptics Symposium*, 2008 pp. 225–230. 1

[2] E. Gobbetti, M. Tuveri, G. Zanetti, and A. Zorcolo. Catheter insertion simulation with co-registered direct volume rendering and haptic feedback. In *MMVR*, 2000 pp. 96–98. 2

[3] R. Klatzky and S. Lederman. Perceiving texture through a probe. In M. L. McLaughlin, J. P. Hespanha, and G. S. Sukhatme, editors, *Touch in virtual environments*, pp. 182–195. 2002. 4

[4] R. L. Klatzky and S. Lederman. Intelligent exploration by the human hand. In S. T. Venkataraman and T. Iberall, editors, *Dextrous Robot Hands*, pp. 66–81. 1990. 1, 2

[5] K. J. Kuchenbecker, J. Fiene, and G. Niemeyer. Event-based haptics and acceleration matching: Portraying and assessing the realism of contact. In *World Haptics Conference*, 2005 pp. 381–387. 1, 2

[6] A. Lecreuse and D. M. Fragaszy. Hand preferences for a haptic searching task by tufted capuchins (*Cebus apella*). *International Journal of Primatology*, 1996 17(4):613–632. 2

[7] S. J. Lederman and R. L. Klatzky. Relative availability of surface and object properties during early haptic processing. *Journal of Experimental Psychology: Human Perception and Performance*, 23(6):1680–1707, 1997. 1

[8] K. E. Overvliet, J. B. J. Smeets, and E. Brenner. Haptic search with finger movements: using more fingers does not necessarily reduce search times. *Experimental Brain Research*, 2007 182(3):427–434. 2

[9] D. K. Pai, K. van den Doel, D. L. James, J. Lang, J. E. Lloyd, J. L. Richmond, and S. H. Yau. Scanning physical interaction behavior of 3D objects. In *SIGGRAPH*, 2001 pp. 87–96. 2

[10] K. A. Purdy, S. J. Lederman, and R. L. Klatzky. Haptic processing of the location of a known property: Does knowing what you've touched tell you where it is? *Canadian Journal of Experimental Psychology*, 58(1):32–45, 2004. 2

[11] G. Robles-De-La-Torre and V. Hayward. Force can overcome object geometry in the perception of shape through active touch. *Nature*, 412:445–448, 2001. 2

[12] K. Salisbury, D. Brock, T. Massie, N. Swarup, and C. Zilles. Haptic rendering: Programming touch interaction with virtual objects. In *Symp. on Inter. 3D Graph.*, 1995 pp. 1054–1059. 3

[13] B. J. Unger, A. Nicolaidis, P. J. Berkelman, A. Thompson, R. L. Klatzky, and R. L. Hollis. Comparison of 3-D haptic peg-in-hole tasks in real and virtual environments. In *IROS*, 2001 pp. 1751–1756. 2

[14] D. C. Wightman and G. Lintern. Part-task training for tracking and manual control. *Human Factors*, 1985 27(3):267–283. 1