MULTIMODAL EXCITATORY INTERFACES WITH AUTOMATIC CONTENT CLASSIFICATION

John Williamson[1] *Roderick Murray-Smith* [1,2]

[1]University of Glasgow
Glasgow
United Kingdom
jhw@dcs.gla.ac.uk

[2]Hamiltion Institute
National University of Ireland, Maynooth
Ireland
rod@dcs.gla.ac.uk

ABSTRACT

We describe an excitation interface for displaying data on mobile devices, based around active exploration: devices are shaken, revealing the contents rattling around inside. This combines sample-based contact sonification with event-playback vibrotactile feedback for a rich and compelling display. Motion is sensed from accelerometers, directly linking the motions of the user to the feedback they receive in a tightly- closed loop. The resulting interface requires no visual attention, and can be operated blindly with a single hand: it is reactive rather than disruptive.

This interaction style is applied to the display of an SMS inbox. We use language models to extract salient features from text messages automatically. The output of this classification process controls the timbre and physical dynamics of the simulated objects. The interface gives a rapid semantic overview of the contents of an inbox, without compromising privacy or interrupting the user.

1. MOTIVATION

We propose a multimodal interaction style where the user *excites* information from a device and then *negotiates* with the system, in a continuous, closed-loop interaction. This draws upon the work of Hermann ([1], [2], [3]), who introduced Model-based Sonification. In [3], the authors state:

> *"...why not sonify data spaces by taking the environmental sound production in our real world as a model. Nature has optimized our auditory senses to extract information from the auditory signal that is produced by our physical environment. Thus the idea is: build a virtual scenario from the data; define a kind of 'virtual physics' that permits vibrational reaction of its elements to external excitations; let the user interactively excite the system and listen."*

In [4] we outline the basis of such a system along with some early prototypes, and discuss a number of interaction scenarios; these include battery life monitoring and display of file system contents. In contrast to non-interactive displays, this *active perception* approach takes advantage of people's expectations about the evolution of dynamic systems. Feedback is tightly coupled to the input. This avoids interrupting or disturbing the user unnecessarily and opens up the potential for richer, more informative feedback. Users know what motions they have made and interpret the display in that context.

Impact perception is a task with which everyone is familiar; few people would have difficulty distinguishing a hollow barrel from a full one after tapping it. Because such information is communicated primarily through the auditory and haptic channels, a completely non-visual interaction can be constructed. Given that mobile devices are often used where visual attention is inconvenient, the use of purely non-visual cues is a major advantage over visually-dominated techniques. By taking a physical model and *overloading* its behaviour, information can be presented very rapidly, without disrupting human intuition about the evolution of physical systems.

The interfaces we have built use inertial sensing for natural motion sensing without any external moving parts; the user just shakes, tilts or wobbles the device to stimulate the auditory and vibrotactile feedback. This can either be an explicit action, or can occur as part of a user's background motion; walking, running, standing up or other everyday motions. This is similar in nature to the "virtual maracas" setup that Fernström proposes in [5] for displaying attributes of a system in an intuitive way. Shaking is a simple but rich motion; a single shake can convey far more information than a simple button press (as might be used in conventional interfaces), and repeated shaking motions are rapid, natural and expressive. The direction, intensity, acceleration profile and timing of motions can be sensed, and it is easy for humans to express themselves in these variables (as babies with rattles quickly learn).

In this paper we demonstrate how automatic classification of text messages can be incorporated into such a system, allowing users to sense rich meta-data *about* the contents of their inboxes in an extremely rapid and natural manner. Each message is represented by a ball free to move within the container. This is combined with a directional filtering technique which provides simple ways of "sieving" the data during the interaction. This could obviously be extended to other collections of textual documents, although scaling the system beyond a few tens of items would require more refined input.

2. BACKGROUND REVIEW

Realistic synthesis of vibrotactile and audio sensations are key to building succesful eyes-free interfaces. There has been a great deal of recent interest in physical models of contact sounds and associated vibration profiles. The model-driven approach is a fruitful design method for creating plausible and interpretable multimodal feedback without extensive *ad hoc* design. Yao and Hayward ([6]), for example, created a convincing sensation of a ball rolling down a hollow tube using an audio and vibrotactile display. A similar sonification of the physical motion of a ball along a beam is de-

scribed in detail in [7]; subjects were able to perceive the ball's motion from the sonification alone. Hermann *et al.* [8] describe an interactive sonification based upon shaking a ball-shaped sensor pack instrumented with accelerometers. This excites data points in a high-dimensional space which are anchored via springs, and produce impact sounds and vibrotactile feedback when they strike each other. The "material" of the objects is used to display the properties of the striking objects. These properties are derived from the geometric properties of the Voronoi tesselation of the data points.

Granular approaches to realistic natural sound generation were explored in [9], where contact events sensed from a contact microphone above a bed of pebbles drove a sample-based granular synthesis engine. A wide variety of sonorities could be generated as a result of physical interactions with the pebbles. This granular approach is used as the synthesis engine in our prototypes. A simple haptic "bouncing ball" on mobile devices was demonstrated by Linjama et al. [10], which used tap sensing to drive the motion of a ball; this, however did not incorporate realistic dynamics or auditory feedback.

The authors discuss granular synthesis based continuous auditory probabilistic displays in [11] and [12], where grain streams are reweighted according to model likelihoods. This work continues the theme of probabilistic display, but does so in an discrete, event-based context, where simplified representations of probability distributions (mode and entropy) are used to characterise the output of multi-model classifiers.

3. INERTIAL SENSING

Motion sensing is achieved by instrumenting mobile devices with tri-axis accelerometers. Accelerometers have previously been widely used for tilting based interfaces (e.g. in [13] and [14]). In the present application, the linear acceleration component is more relevant than gravitational effects; the tilting is of less consequence than deliberate shaking motions. Accelerometer inputs are highpass filtered (see Section 4) to eliminate the effects of slow tilt changes.

Prototypes of this system run on iPaq 5550 devices (see Figure 1), using the MESH ([15]) device for inertial sensing and onboard vibrotactile display. The MESH's vibrotactile transducer is a VBW32 loudspeaker-style device. This device allows for the display of high fidelity tactile sensations due to its large bandwidth and fast transient response. This is used in combination with the iPaq's internal eccentric motor vibration unit, providing a lower frequency range of vibration sensations than achievable with the VBW32, like a woofer would in an audio system, at the cost of reduced temporal resolution. The accelerometers are used for sensing and are sampled at 100Hz, with a range of approximately $+/-$ 2g. The gyroscopes and magnetometers are not used. Capacitive sensing is used for tap detection (see Section 5.2.2).

Earlier versions of the "Shoogle" system [4] also run on standard mobile phones (such as the Nokia Series 60 devices), using the Bluetooth SHAKE inertial sensor pack (see Figure 2). This provides accelerometer measurement and vibrotactile feedback in a matchbox size wireless package, along with a range of other sensing functionality (gyroscopes, magnetometers and capacitive sensing). Work is underway in porting the automatic text message classification system to work transparently with real SMS inboxes on mobile phones, so that the system can be tested "in the wild".

Figure 1: *The MESH expansion pack, with an iPaq 5550 PocketPC. This provides accelerometer, gyroscope and magnetometer readings, as well as vibrotactile display.*

Figure 2: *The wireless SHAKE sensor, shown with a 2 Euro piece and a Nokia 6660 for size comparison. This Bluetooth device comprises a complete inertial sensing platform with onboard vibrotactile feedback.*

4. OBJECT DYNAMICS

The behaviour of the interface is governed by the internal physical model which defines the relation between sensed motions and the response of the component objects whose interactions generate feedback events. Here, the simulated physical system is based around objects bouncing around within a rectangular box whose physical dimensions *appear* to be the same as the true physical dimensions of the device. Each message in the inbox is mapped onto a single spherical object.

4.1. Accelerometer Mapping

The accelerations sensed by the accelerometers are used directly in an Euler integration model. This is quite sufficient given the relatively fast update rate and the nonstiffness of the dynamics; the feedback properties also mean than small numerical innacuracies are impercetible. The accelerations are highpass filtered to remove components under $\sim$0.5Hz. This eliminates drifts due to changes in orientation, and avoids objects becoming "stuck" along an edge. These accelerations are then transformed by an object-dependent rotation matrix (based on the message sender, see Section 5.2.1) and a scaling matrix which biases the objects along a specific axis:

$$\ddot{x}_i = [a_x\ a_y\ a_z] R_x(\theta_i) R_y(\phi_i) R_z(\psi_i) \begin{bmatrix} 1 \\ \alpha \\ 0 \end{bmatrix}, \qquad (1)$$

where θ_i, ϕ_i and ψ_i are the rotation angles for object i, and $0 < \alpha < 1$ is the direction-biasing term (see Figure 3). Any other projection of the acceleration vector could be used; this could include

Figure 3: *The accelerations are transformed before application to the objects, such that each object "class" is excitable in a different direction. The cylinders here illustrate the different directions along which the device can be moved to excite different classes (although these are rather more oval shaped in the simulation). The cut-aways show the virtual objects (ball-bearings) within.*

more complex nonlinear projections. For example, the feature vector could be extended to include a quadratic basis, while still using a simple linear projection matrix. The vector could also include estimated time-derivatives of the acceleration measurements.

4.2. Friction and Stiction

Nonlinear frictional damping is applied to the motion of the objects. This eliminates rapid, small, irritating impacts caused by slight movements, while remaining realistic and sensitive. The friction function is a piecewise constant function, so that:

$$f = \begin{cases} f_s & (|\dot{x}| < v_c) \\ f_m & (|\dot{x}| \geq v_c) \end{cases}, \tag{2}$$

where f_s and f_m are the static and moving coefficients and v_c is the crossover velocity. These coefficients can be varied to simulate different lining materials within the box, or different object surfaces (e.g. smooth plastic versus velvet balls).

4.3. Springs

When objects are created, they are attached to a randomly-allocated position within the simulated box by a linear Hooke-law spring, such that:

$$\ddot{x}_q = a_q + \frac{k(x_q - x_{q0})}{m}, \tag{3}$$

where x_{q0} is the anchor point (see Figure 4). The spring coefficient k loosens or tightens the motion of the balls. Without this spring motion, the system can feel unresponsive as objects tend to cluster together as a consequence of the constraints created by walls. It

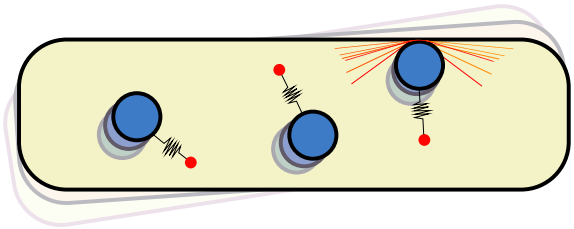

Figure 4: *The simulated system. A number of balls, anchored via springs, bounce around within the virtual container. When they impact (as in the top right) sound and vibration are generated, based on the physical properties of the impact and the properties of the message with which they are associated.*

is only through the spring force that the mass of the balls enters the dynamics calculations, although a more realistic system could include a mass term in the friction computation.

4.4. Impacts

Feedback events are generated only when the balls collide with the walls of the device "box". These impacts trigger sample playback on both the vibrotactile and audio devices. Inter-ball collisions are not tested for. Wall collisions are inelastic, transferring some kinetic energy to the wall, and the remainder to rebound. The rebound includes directional jitter – simulating a slightly rough surface – to reduce repetitive bouncing.

5. MESSAGE TRANSFORMATION

Each of these impacts is intended to communicate information about the message with which it is associated. Due to the limitations of the haptic transducers, the vibrotactile feedback varies in a very limited way; it largely serves as an indicator of presence and gives an added degree of realism. The properties of each message are instead sonified, modulating the impact sounds to reveal meta-data. This meta-data is intended to summarise the contents of the inbox in a way which maintains user privacy (other listeners will not obtain significant personal information) and can be presented extremely rapidly.

Several transformations are used in the sonification process. The simplest transformation involves linking the mass of each ball to the length of the SMS message. Longer messages result in heavier balls with appropriate dynamics, and suitably adjusted resonances. The most important feature is association of impact "material" to the output of a classifier which identifies language styles within the message. This is similar in nature to the language model based sonifications used in the speed-dependent automatic zooming described in [16]. This aims to give an interacting user some sense of the content or style of the messages rapidly, without visual display or laborious text-to-speech output (which also has obvious privacy issues). The identity of the message sender and relative time of arrival of the messages are also displayed, by directional filtering and a rhythmic display, respectively. The impact sounds are also panned according to the site of the impact; however this is only useful when the device is used with headphones.

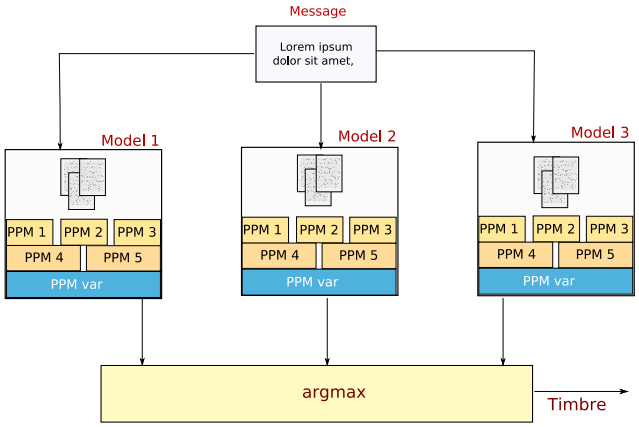

Figure 5: *The structure of the text classifier. Multiple models are run on incoming messages. Each of these models is composed of a number of weighted submodels. The index of the classifier with highest posterior likelihood is used to select the impact timbre.*

5.1. PPM Language Model

The language modeling involves multiple partial-predictive-match (PPM) models ([17], [18]). These code text very well, approaching the theoretical maximum compression for English texts (see [19]). Figure 5 gives an overview of the process. Each class of messages has a separate model θ_j, and is trained on a corpus of messages in that style (e.g. in a particular langauge, or in a specific vernacular).

The current implementation uses a hybrid structure, with one submodel trained with variable length prefix, running from the start of the word and terminating at the first non-letter character (which is particularly sensitive to keywords), combined with a smoothed fixed-length PPM submodel. This provides a measure of robustness in classification, especially where training texts are sparse.

The smoothed submodel combines weighted PPM models of different length (designated PPM_h for a length h model, with PPM_0 being the model trained with no prefix, i.e. the independent distribution of characters), up to a length 5 model:

$$p(c|r_f) = \sum_{h=0}^{5} \lambda_h p(c|\text{PPM}_h) \tag{4}$$

with the $\lambda_0 \ldots \lambda_5$, $\sum_{h=0}^{5} \lambda_h = 1$ determining the weighting of the models. Symbols unseen in the training text are assigned a fixed probability of $\frac{1}{S}$, where S is the total number of possible symbols (e.g. all ASCII characters).

The fixed length and word length classifiers are used in a weighted combination, such that under each model θ_j, each character has a probability

$$p(c|\theta_j) = \lambda_q p(c|r_v) + (1 - \lambda_q) p(c|r_f), \tag{5}$$

where λ_q is a weighting parameter, r_v is the variable length model and r_f is the smoothed fixed length model.

This smoothing ensures that when the variable length model abruptly fails to predict subsequent characters (e.g. when a word not seen in the training text appears), the system relaxes back to the fixed-length model, and the fixed-length model will, in the worst case, revert to the independent distribution of characters in the training text. For each model, a tree of probabilities is stored, giving $p(c|r)$, where c is the character predicted, and r is the current prefix. The variable length model is pruned during training so that nodes with observation count below a certain threshold are cut off after a specified number of symbols have been seen. This reduces the size of the tree sufficiently that it can be used without excessive memory consumption. A section of an example tree is given in Figure 6.

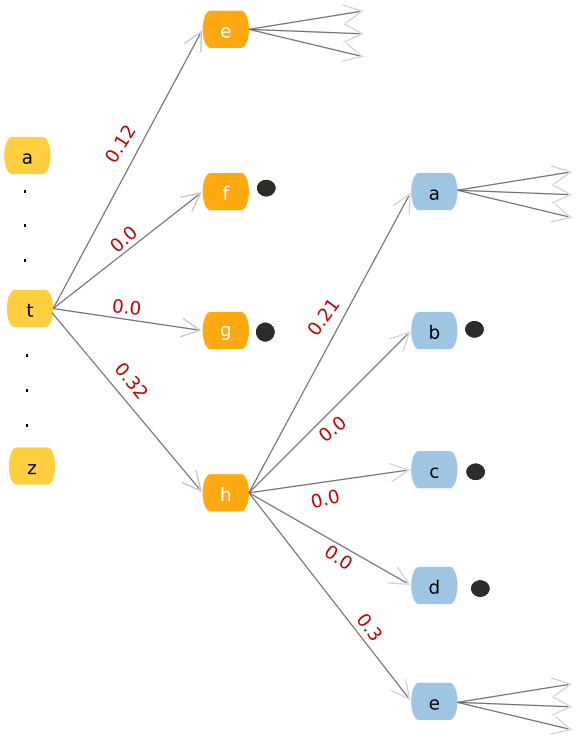

Figure 6: *A sample language model tree. The model is stored as a graph, with the edge weights being the probabilities of each transition, obtained from the normalized counts of that transition from the training corpus. Solid circles indicate no further transitions from this symbol. Each submodel has its own tree.*

The likelihood of a message is then just

$$p(\theta_j|\text{text}) = \frac{p(\text{text}|\theta_j)p(\theta_j)}{p(\text{text})}. \tag{6}$$

Assuming constant prior across all all models, and under the assumption that any message must belong to one of the initially trained classes, the model probabilities can be normalized such that $\sum_{k=0}^{n} p(\text{text}|\theta_k) = 1$, i.e. the posterior likelihood becomes just

$$p(\theta_j|\text{text}) = \frac{p(\text{text}|\theta_j)}{\sum_{k=0}^{n} p(\text{text}|\theta_k)}. \tag{7}$$

For each model, we have

$$\log p(\text{text}|\theta_j) = \sum_{i=0}^{\text{END}} \log p(c_i|\theta_j). \tag{8}$$

Exponentiating and substituting (8) into (7), the model likelihoods are obtained.

Name	Corpus	Model
SMS	SMS messages (Singapore corpus [20])	θ_1
Finnish	Finnish poetry (Project Gutenberg)	θ_2
News	Collection of BBC News articles	θ_3
German	German literature (Project Gutenberg)	θ_4
Biblical	Old testament (KJV Genesis)	θ_5

Table 1: *Test classes and the training texts used for them.*

5.1.1. Potential Enhancements

Although word level models of text could be introduced, these have significant training requirements and require enormous storage for anything but the smallest corpora. Given the excellent compression capabilites of the PPM models and the good performance of the classifiers in the test scenarios (see Section 5.1.2) additional modelling is probably excessive for the current problem.

Ideally, each of these classes would be adapted online, with the user assigning incoming messages to particular classes, to cope with the various styles that user regularly receives. Although this is not currently implemented, it would be relatively simple to do so.

5.1.2. Test Model Classes

For testing, five language classes were created by selecting appropriate corpora. Due to the lack of suitable text message corpora in different styles, a number of artificial classes were created. Although these differences are exaggerated compared to the types of messages commonly received, they demonstrate the utility of the techinque. Each of these was trained with a relatively small corpus, but this was sufficient given the very significant differences in language. Table 1 shows the test classes and their corresponding corpora.

Table 2 shows testing results from these classifiers. The table shows example messages and their "true" class, along with the classifier probabilities and the classifier entropy. The classifier performs extremely well, even for the less well distinguished categories.

5.1.3. Certainty Filtering

The timbre of the impact is always set to be the one associated with the most likely model (see Section 6.2), but the output of the classifiers is clearly uncertain; message classes can often be quite similar, and many messages may be either so short or so generic as to be indiscriminable. The system displays this uncertainty by manipulating the audio according to the entropy of the classifier distribution

$$H(\theta|\text{text}) = -\sum_{j=0}^{n} p(\theta_j|\text{text}) \log_2 p(\theta_j|\text{text}). \qquad (9)$$

This is entropy sonification is performed in two ways: firstly, when $H(\theta|\text{text})$ rises above some threshold $H_{\min}$, the timbre associated with the message is set to be a special class representing a "general message"; secondly, when the entropy is below this threshold, a lowpass filter is applied to impact sound, with the cutoff inversely proportional to the entropy:

$$c = \frac{z}{\epsilon + H(\theta|\text{text})}, \qquad (10)$$

for some constants z, ϵ. This dulls the sound as the entropy increases.

5.2. Exploration

The basic shaking system simply gives a sense of the numerosity and composition of the messages in the inbox. Two additional features extend the interaction to present other aspects of message content. These rely on the device being stimulated in different ways, rather than attempting to present more information in a single impact.

5.2.1. Identity Sieving

Identity sieving links the sender or sender group (e.g. family, friends, work) of the message (which can be obtained from the message meta-data) to a particular plane in space (see Section 4). These messages can be excited by moving the device in this plane; moving it in others will have less effect. A user can "sieve out" messages from a particular sender by shaking in various directions. The stiction model increases the selectivity of this sieving, so that objects who are free to move along other plains tend not to do so unless more violent motions are made. All objects can still be excited by making such vigorous movements.

5.2.2. Time-sequenced "Rain"

An additional overview of the contents of the inbox can be obtained by tapping the device, causing the balls to shoot "upwards" (out of the screen), and then fall back down in a structured manner. Tapping is sensed independently using the capacitive sensors on the MESH device, and is reliable even for gentle touches while being limited to a small sensitive area. The falling of the objects is linked to the arrival time of the messages in the inbox, with delays before impact proportional to the time gap between the arrival of messages. The rhythm of the sounds communicates the temporal structure of the inbox. Figure 7 illustrates this.

6. AUDITORY AND VIBROTACTILE DISPLAY

The presentation of timely haptic responses greatly improves the sensation of a true object bouncing around within the device over an audio-only display. As Kuchenbecker *et al* [21] describe, event-based playback of high-frequency waveforms can greatly enhance the sensation of stiffness in force-feedback applications. In mobile scenarios, where kinaesthetic feedback is impractical, event-triggered vibration patterns can produce realistic impressions of contacts when the masses involved are sufficiently small.

6.1. Vibrotactile Events

The vibrotactile waveforms sent to the VBW32 transducer on the MESH device are enveloped sine waves, with a frequency of 250Hz. This is at both the resonant frequency of the transducer, and around the peak sensitivity of the skin receptors involved in vibrotactile perception.

Text	**True Class**	$\log_2 p(\theta_1)$	$\log_2 p(\theta_2)$	$\log_2 p(\theta_3)$	$\log_2 p(\theta_4)$	$\log_2 p(\theta_5)$	$H(\theta)$
"yo what up wit u the night. u going to the cinema?"	SMS (θ_1)	-4.23×10^{-5}	-116.47	-18.98	-128.82	-15.15	$\mathbf{4.9 \times 10^{-4}}$
"Ken aina kaunis on, ei koskaan pöyhkä, Nopea kieleltänsä, mut ei röyhkä; Ken rik as on, mut kultiaan ei näytä; Tahtonsa saada voi, mut sit ei käytä."	Finnish (θ_2)	-233.32	0.0	-205.36	-186.36	-224.15	$\mathbf{1.47 \times 10^{-54}}$
"They are charged with crimes against humanity over a campaign against Kurds in the 1980s."	News (θ_3)	-80.08	-233.69	0.0	-253.54	-77.39	$\mathbf{4.51 \times 10^{-22}}$
"Die Erde ist nicht genug, Mond und Mars offenbar auch nicht: Google will demnächst das gesamte Universum erfassen."	German (θ_4)	-185.69	-213.26	-159.05	0.0	-178.71	$\mathbf{2.09 \times 10^{-46}}$
"Now after the death of Joshua it came to pass, that the children of Israel asked the LORD, saying, Who shall go up for us against the Canaanites first, to fight against them."	Biblical (θ_5)	-126.82	-438.91	-80.56	-484.06	0.0	$\mathbf{4.99 \times 10^{-23}}$

Table 2: *Test results with the five classifiers. These short texts were taken from documents not present in the training corpus. Zeros are shown where the probability is so close to 1 that the logarithm is not computed exactly. The entry with maximum probability for each row is highlighted in light grey.*

Several basic envelope shapes are pre-programmed: simple impacts with very rapid linear attack, and an exponential decay; sloshing-like vibrations with much longer attack portions, and slightly varying frequencies; granular effects with sums of extremely short enveloped sine waves at random time delays; and heavier impacts with long ringing portions and initially saturated output. Figure 8 gives an overview of these waveforms.

To enhance the sensations, the vibrations from internal eccentric motor of the iPaq is layered with the VBW32 transducer. The short, high-frequency vibrations are sent to the smaller vibrator, with heavy, slow impacts routed to the motor-driven actuator. The greater power of the motor-driven actuator results in a more "solid" feel, but limited control restricts output to simple events played in conjunction the high-frequency events (similar to the layering of sub-bass waveforms under Foley effects in film sound design).

Even when the inbox is empty, gentle vibration feedback is produced by the internal vibration motor in response to movement to indicate that the system is "live" and sensing motion.

6.2. Audio Synthesis

The impact of the balls on the virtual box produces sound related to the physics of the collisions. Although ideally these sounds would be generated by a physical model (such as the general contact sound engine given by van den Doel *et al* [22]), the limited computational power of many mobile devices – which lack efficient floating-point units – makes this difficult. Such techniques, would, however, greatly increase the potential expressivity of the interface.

6.2.1. Sample Banks

Given these computational restrictions, the system instead relies on a granular technique, with wavetable playback for synthesis, combined with some simple signal post-processing. This gives a high degree of realism, but comes at the cost of significant effort in creating a sample library, and limited flexibility.

A number of impact sounds (8–16) are pre-recorded for each of a number of impact types (wood on glass, for example). These slight variations are critical to avoid artificial sounding effects. On impact, a random sound from within this class is selected and mixed into the output stream. The audio output – which uses the FMOD library – mixes up to thirty-two simultaneous channels, to ensure that impacts do not cut off previous audio events in an unnatural manner.

A listing of some of the impact types which have been sampled is given in Table 3. These provide a wide range of natural sounding and easily distinguishable timbres, although the limited quality of the speakers on the iPaq introduces some unnatural artifacts. Humans are exceedingly adept at inferring the physical properties of materials from the sound of their physical interaction, and the realistic nature of the generated sounds makes the nature of the impact immediately obvious.

Due to technical limitations of the implementation hardware, synchronisation of audio with the vibrotatile is limited, with variable delays of up to 50ms between the modalities. This does not, however, seem to reduce the subjective realism or quality of the interaction.

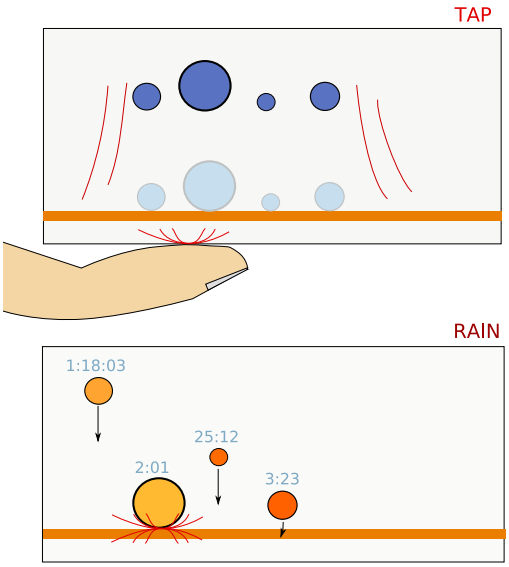

Figure 7: *Capacitive sensors detect tap motions on the case of the device. This causes the virtual objects to be launched up, and then rain back down so that the timing of the impacts reveals the relative timing of the arrival of the associated messages. Here, the times shown in light blue are arrival times (the time since this message arrived), so that the newest message impacts first.*

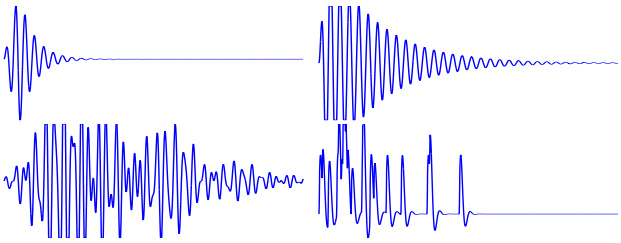

Figure 8: *Four different vibrotactile waveform types. Left-to-right, top-to-bottom: standard "light ball" impact; heavy, ringing impact; liquid sloshing; gritty, particulate impact. All of these have energy concentrated around the 250Hz band.*

6.2.2. *Audio Transformations*

The samples are transformed based on the properties of each particular collision, to communicate as much information as possible given the limited processing capabilites and limited sample set. The gain of the impact sound is set to be proportional to the kinetic energy of the impact ($\frac{1}{2}mv^2$); this is essential for natural sounding effects. As described in Section 5.1.3, the entropy of the language class distribution is mapped to the cutoff of a simple one-pole IIR filter running on the impact sound, so that less certain classifications have a duller tone. The sample is also pitch-shifted with a simple resampling process in proportion to the mass of the ball which impacted, so that large-mass interactions produce lower sounds with deeper resonances than smaller ones. This transformation can only reasonably be applied to sound types which have clear resonances *in the object which impacts* – the effect is quite unnatural for sounds such as sloshing water, where pitch-shifted audio sounds like liquids of different viscousity in differently sized containers. Figure 9 gives an overview of the synthesis process.

Ideally there would be impulse responses for both the object and the container which would be convolved at runtime; pitch-shifting the object impact could independently change its perceived mass without interfering with the formants of the container impulse. However, this is computationally challenging with the current generation of mobile devices.

Name	Description
pingpong	Ping pong balls dropped onto a wooden board
anvil	Miniature anvil struck with miniature hammer
glass	Ordinary glasses struck with a wooden rod
chime	Small metal chimes
jar	Small hard candies hitting a glass jar
water	Water drops in a cup
slosh	Water sloshing in a glass
bigslosh	Water sloshing in a large demijohn
gravel	Gravel hitting gravel
tick	Mechanical clock ticking
klang	Long metal bar being struck
metal	Metal radiator struck with a metal bar
bubble	Bubbles formed by air blown in to water
viscousbloop	Stones being dropped into a very viscous liquid
keys	Keys jangling in a pocket
didj	A didjeridoo being struck with a hand

Table 3: *A listing of some of the impact classes which have been sampled. Each of these impacts sampled numerous times so that the natural variations in the sounds are retained.*

7. FUTHER WORK – ACTIVE SELECTION

The current prototype has elementary "sieving" functions for sensing particular aspects of data within the device. More sophisticated selection methods, such as the active selection techinques described in [23], could be used to identify and select individual messages within the collection. This could involve modality scheduling, where the interaction moves from simple vibrotactile information (the presence of messages), through auditory display (the composition of the collection) and finally to visual display, for on-screen display of messages. Alternatively, speech synthesis could be applied in the final stage for an entirely non-visual message browsing system. Such a system would be a significant step towards a usable non-visual, buttonless mobile platform.

8. CONCLUSIONS

This system extends our previous work on excitatory interfaces to include the sonifications based on the automatic classification of messages, as well as introducing the "sieving" metaphor to extend the display capabilites of the system. The automatic classification process could easily be extended to other data forms; music files could be linked to objects with properties obtained from the automatic classification of genres, for example.

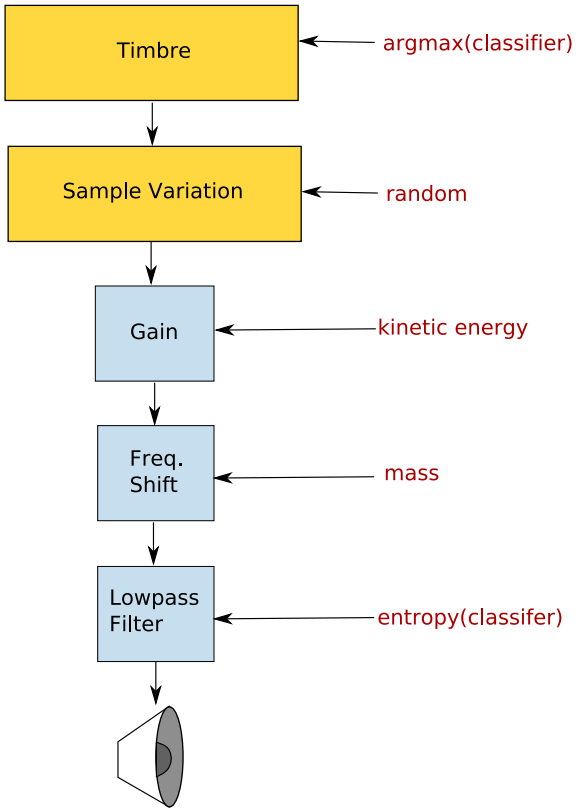

Figure 9: *The synthesis process. A wave is selected from the wavetable, and then transformed according to the object properties.*

The model-based interaction leads to an intuitive and compelling interface which is well suited to the physical manipulability of mobile devices. The metaphor of the mobile device as a physical container within which interaction can take place is one that can be extended to many other interaction scenarios. The interface is based on active sensing, where users drive the interaction at their own pace; the system does not interrupt of its own accord. The result is a rich multimodal display that can be used without visual attention, taking advantage of user's familiarity with the dynamics of processes in the physical world to present information in a natural and non-irritating manner.

9. ACKNOWLEDGEMENTS

The authors are grateful for support from: the IST Programme of the European Commission, under PASCAL Network of Excellence, IST 2002-506778; the IRCSET BRG project BRG SC/2003/271 Continuous Gestural Interaction with Mobile devices; HEA project Body Space; and SFI grant 00/PI.1/C067. This publication only reflects the views of the authors.

Audio examples and a video are available online at `www.dcs.gla.ac.uk/~jhw/shoogle/`

10. REFERENCES

[1] T. Hermann, *Sonification for Exploratory Data Analysis*, Ph.D. thesis, Bielefeld University, Bielefeld, Germany, 2002.

[2] T. Hermann and H. Ritter, "Crystallization sonification of high-dimensional datasets," *ACM Transactions on Applied Perception*, vol. 2, no. 4, pp. 550–558, 2005.

[3] T. Hermann and H. Ritter, "Listen to your data: Model-based sonification for data analysis," in *Advances in intelligent computing and mulimedia systems.*, M. R. Syed, Ed., pp. 189–194. Int. Inst. for Advanced Studies in System Research and Cybernetics,, 1999.

[4] J. Williamson, R. Murray-Smith, and S. Hughes, "Shoogle: Excitatory multimodal interaction on mobile devices," in *Proceedings of CHI 2007*, 2007, p. In Press.

[5] M. Fernström, "Sound objects and human-computer interaction design," in *The Sounding Object*, D. Rocchesso and F. Fontana, Eds., pp. 45–59. Mondo Estremo Publishing, 2003.

[6] H.-Y. Yao and V. Hayward, "An experiment on length perception with a virtual rolling stone," in *Eurohaptics 06*, 2006.

[7] M. Rath and D. Rocchesso, "Continuous sonic feedback from a rolling ball," *IEEE MultiMedia*, vol. 12, no. 2, pp. 60–69, 2005.

[8] T. Hermann, J. Krause, and H. Ritter, "Real-time control of sonification models with an audio-haptic interface," in *Proceedings of the International Conference on Auditory Display*, R. Nakatsu and H. Kawahara, Eds., Kyoto, Japan, 7 2002, International Community for Auditory Display (ICAD), pp. 82–86, ICAD.

[9] S. O'Modhrain and G. Essl, "Pebblebox and crumblebag: Tactile interfaces for granular synthesis," in *NIME'04*, 2004.

[10] J. Linjama, J. Hakkila, and S. Ronkainen, "Gesture interfaces for mobile devices - minimalist approach for haptic interaction," in *CHI Workshop: Hands on Haptics: Exploring Non-Visual Visualisation Using the Sense of Touch*, 2005.

[11] J. Williamson and R. Murray-Smith, "Sonification of probabilistic feedback through granular synthesis," *IEEE Multimedia*, vol. 12, no. 2, pp. 45–52, 2005.

[12] J. Williamson and R. Murray-Smith, "Granular synthesis for display of time-varying probability densities,," in *International Workshop on Interactive Sonification*, A. Hunt and Th. Hermann, Eds., 2004.

[13] J. Rekimoto, "Tilting operations for small screen interfaces," in *ACM Symposium on User Interface Software and Technology*, 1996, pp. 167–168.

[14] K. Hinckley, J. Pierce, M. Sinclair, and E. Horvitz, "Sensing techinques for mobile interaction," in *UIST'2000*, 2000.

[15] S. Hughes, I. Oakley, and S. O'Modhrain, "Mesh: Supporting mobile multimodal interfaces," in *UIST 2004*. 2004, ACM.

[16] P. Eslambolchilar and R. Murray-Smith, "Model-based, multimodal interaction in document browsing," in *Multimodal Interaction and Related Machine Learning Algorithms*, 2006.

[17] T. Bell, J. Cleary, and I. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Transactions on Communications*, vol. 32, no. 4, pp. 396–402, 1984.

[18] J. Cleary, W. Teahan, and I. Witten, "Unbounded length contexts for PPM," in *DCC-95*. 1995, pp. 52–61, IEEE Computer Society Press.

[19] W. J. Teahan and J. G. Cleary, "The entropy of English using PPM-based models," in *Data Compression Conference*, 1996, pp. 53–62.

[20] Y. How and M.-Y. Kan, "Optimizing predictive text entry for short message service on mobile phones," in *Human Computer Interfaces International*, 2005.

[21] K. J. Kuchenbecker, J. Fiene, and G. Niemeyer, "Improving contact realism through event-based haptic feedback," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 2, pp. 219–230, 2006.

[22] K. van den Doel, P. G. Kry, and D. K. Pai, "Foleyautomatic: physically-based sound effects for interactive simulation and animation," in *SIGGRAPH '01*. 2001, pp. 537–544, ACM Press.

[23] J. Williamson, *Continuous Uncertain Interaction*, Ph.D. thesis, University of Glasgow, 2006.