(Preprint. To be published in the proceedings of the "Touch, Blindness, and Neuroscience.", Madrid, Spain, 2002/10/16-18.)

RENDERING OF MOON TEXT ON SIMULATED TACTILE DIAGRAMS FOR BLIND COMPUTER USERS BY FORCE-FEEDBACK

Andrew John Hardwick
BTexact Technologies, Adastral Park, Ipswich, IP5 3RE, UK.
E'mail: andrew.hardwick@bt.com .

1. SUMMARY

This paper describes the first computer system for dynamically displaying text in Moon, a tactile script for blind users, using force-feedback simulation.

The Moon alphabet is a standard alternative to the better known Braille alphabet in which letters consist of embossed lines, rather than dots. It is considered to be easier for elderly blind people to learn but the uptake has been severely hindered by the lack of a computer display method, other than printing or embossing it onto paper, whereas many electromechanical Braille displays are available. The invention reported in this paper is a solution to that problem.

2. BACKGROUND

The Moon text computer display system was originally invented for labelling tactile diagrams. The tactile diagrams blind people normally use are mechanically or thermally embossed on sheets of paper or plastic. Whilst good for many tasks, they are slow to produce, bulky to store and not suitable for dynamically changing diagrams. For those tasks, an obvious potential solution is to use a computer haptic (tactile) output device to simulate the feel of a tactile diagram as a computer monitor simulates the appearance of printed visual diagrams.

The most suitable haptic output devices currently available are 'force-feedback' displays which are typically small robot arms. The user holds and moves the arm, the device senses its position, the computer calculates a suitable reaction force to simulate touching an object, and the device applies that force to the user. However, it is difficult to label the diagrams using such a device. Real tactile diagrams are labelled in Braille and the 'vBraille' system (Petrie 1999) attempted to replicate that on a force-feedback display. However, current mechanical limitations restrict force-feedback devices to, at most, one contact point per finger so it is similar to feeling the text with a stylus or whilst wearing a hard thimble. Braille is designed to be read a character at a time by feeling multiple dots on a finger tip simultaneously not to be traced out with a pin. Feeling Braille thus requires considerable concentration to build-up a mental image of a character from its dots and is tediously slow.

One solution would be to use a dot-matrix haptic display, as is used for normal computer Braille output, instead a force-feedback one. However, their mechanical limitations do not allow as good resolution & variable heights for diagrams. The alternative solution described is to use a force-feedback display but to use Moon instead of Braille for the text.

3. MOON TEXT

Moon text was invented in 1845 by Dr. William Moon (RNIB 2002) and is now overseen by the Royal National Institute for the Blind. Whereas Braille uses dot patterns unrelated to a visual alphabet, Moon is composed of curved lines and is partly derived from the Roman alphabet (Figure 1).

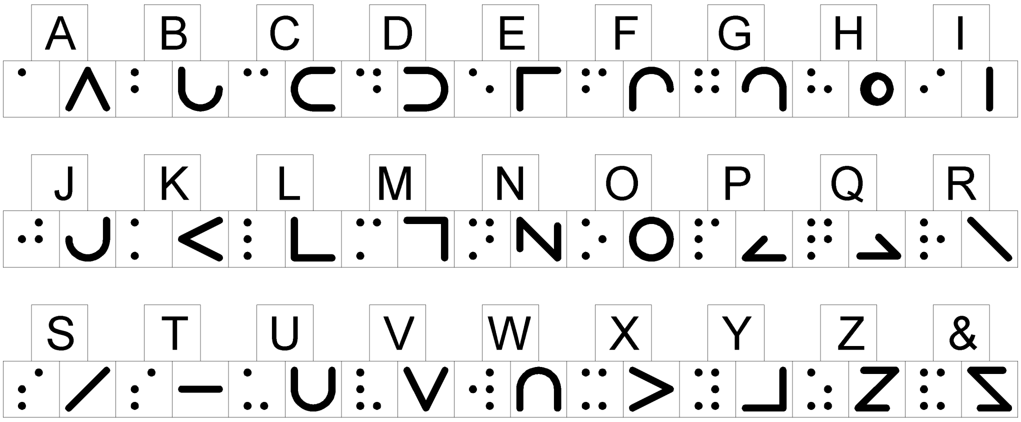

Figure 1: The basic Braille (left) and Moon (right) tactile scripts.

Braille is optimised for speed of reading by experienced users whereas Moon is optimised for speed of learning by new users familiar with the printed alphabet. However, the crucial advantage Moon text has for force-feedback display is that its lines can be traced out easily, especially if it is converted from an embossed ridge to an engraved groove (Figure 2).

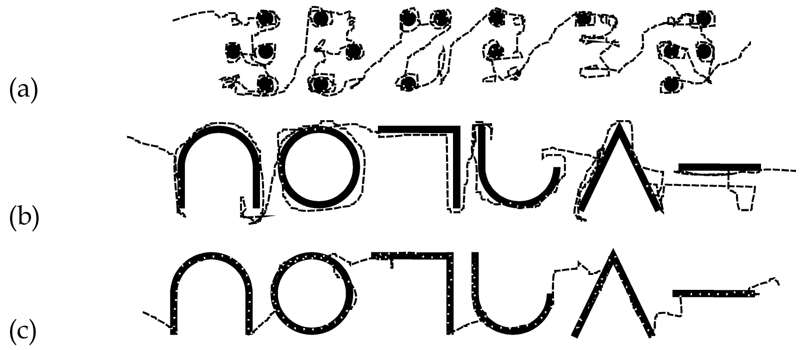

Figure 2: Illustrative paths for tracing out variation tactile scripts using a point contact: (a) embossed Braille, (b) embossed Moon, (c) engraved Moon.

The Moon 'H' is an anomalous character, being the only one dependent upon size or line width for identification. To avoid the necessity of users checking for these factors, I used a more homologous replacement shaped like a reverse 'N' (a Cyrillic 'I'). Which to use could be left as an option for individual users.

4. IMPLEMENTATION

The system was implemented by adapting an existing system for simulating textured objects as force-fields on force-feedback displays (Hardwick *et al* 1998) but replacing the periodic rough textures with aperiodic textures in the shape of grooves or ridges which form the shape of Moon characters.

Textures in this system were represented as 2-dimensional maps of displacement & direction. These maps could be based on 2 dimensional arrays of texture data but, in this case, were functions which analytically calculated the texture data for any position as needed and thereby allowed arbitrarily high precision and ensured no artefacts, such as faceting of curves, from discrete sampling. The displacement returned by the function was the height, above the surface plane, which the texture locally adds to a flat surface to comprise the required textured surface. Indentations were simply treated as negative heights. The direction is the unit vector normal to the local surface slope at that location.

The textures used for Moon characters were triangular grooves (or ridges). The edges were slightly rounded to eliminate the instability which can arise from the force-feedback device oscillating across a narrow region in which a force component reverses between sampling periods. Two textures of this form were created, a straight line specified by the coordinates of its endpoints and circular line specified by it centre's coordinates and its radius. The circular line texture could be modified into two other textures, a circular arc and a single dot, by restricting its angular extent and setting its radius to zero respectively. The line depth (or height) and width of these textures were also alterable parameters.

Objects were represented as 3-dimensional force-field maps giving the force to output via the force-feedback device for each position. The force for each point within each object was a normal reaction directed towards the nearest external surface. Outside all objects, the force was zero. The magnitude of the force inside an object started from zero at its surface and increased linearly with depth below the surface (i.e. Hooke's spring law) to feel like a elastic surface and to minimise the instabilities which a force-field discontinuity could cause. It was capped at a value below the mechanical limit of the output hardware. The object in this case, representing the paper surface, was just a trivial flat horizontal plane.

Textures were applied to objects by their displacements being used to locally modify the calculated depths used in determining whether or not at a given point was inside an object and, if so, what magnitude of force to generate. The directions of those forces were then modified using the local normal direction vector from the texture before being output.

Characters were composed from the 2 fundamental & 2 derived elementary textures by combination in texture groups. Texture groups behaved just like other textures except that their parameters were a collection of subsidiary textures (which could include other texture groups). They calculated the combination of the multiple textures by returning the displacement and normal direction from the subsidiary texture with the greatest absolute displacement at the required position, a simple procedure which mitres joints correctly for both ridges & grooves. The straight, circular & arc line textures were used in texture groups to compose character textures and the dot texture was added to smoothly cap line texture endings and seamlessly round off the corners where straight lines met (Figure 3).

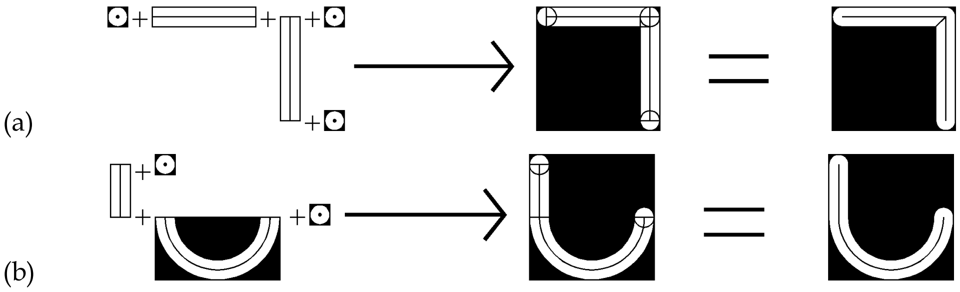

Figure 3: Composition of Moon character textures, (a) 'M' & (b) 'B'. (Left) Elementary textures. (Middle) Combined textures forming the letter shapes. (Right) Combined textures showing smooth joins resulting from the dot corner pieces and greatest absolute displacement texture combination method. The solid black areas are bounding rectangles where they extend beyond the basic textures.

The same method which by texture elements were combined into character textures was repeated to combine character textures hierarchically into lines texture and line textures into a page texture (Figure 4). Texture groups maintained records of the bounding rectangles at each stage which enabled the particular element being touched at any time to be quickly found without calculating every element of every character on the page.

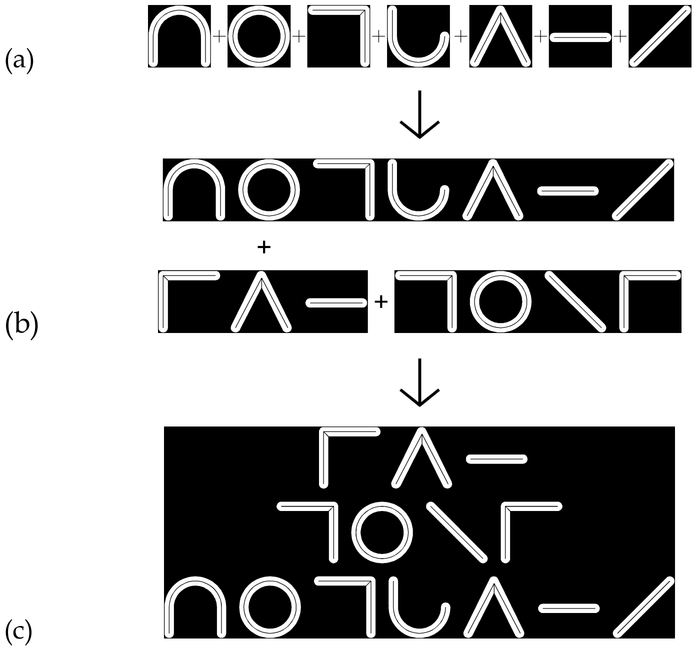

Figure 4: Further hierarchical combination of letter textures, (a), from Figure 3 into line textures, (b), into a page texture, (c), showing the successive nested bounding rectangles (solid black).

The system was programmed in C++ with care taken to optimise for speed in the parts which were to be executed in the time-critical force-feedback loop whilst maintaining a clear robust program architecture for future extensibility. C was chosen for speed and augmented to C++ because texture elements, basic objects and hierarchical groupings naturally fit an Object Oriented Programming structure. The fundamental parts of the program were not computer, operating system or force-feedback hardware specific although, of course, for actual compilation & use, isolated replaceable parts were inserted to interface it to the particular platform in use.

5. RESULTS

The result was quite efficient: when implemented in C++ on Microsoft™ Windows NT4 a 400 MHz Pentium™ II Dell™ PC, it calculated reaction forces (averaged over 10^4 positions) from single characters in 0.14 to 38 μs, depending upon the character. It also scaled well, doing a page of representative text, 64 x 30 characters in size, $1.5 \times 10^4 \pm 1 \times 10^3$ times per second.

The C++ program was used as a Dynamic Link Library and given two alternative user interfaces – a graphical user interface written in Labview for showy demonstrations and a text-based one written in Perl more appropriate to blind users. A Sensable™ Phantom™ force-feedback device with the simulation force-feedback loop running at 1000 kHz was used as the output hardware.

We have yet to do formal scientific trials with system but initial reactions by the blind people who have felt it have been very favourable and the simulation is so clear that most of sighted people who have tried have had no problem tracing out characters despite their relative lack of experience. The only problem found was that some users put too much weight on the simulated page which overheated the Phantom™'s motors as it tried to maintain sufficient reaction force. Sloping the page towards the user discouraged that.

6. DISCUSSION

Although it was primarily intended for labelling computer-simulated tactile diagrams, the system can also display Moon text for general reading tasks which, at last, gives the possibility of dynamic computer Moon output of digital texts including Web pages and 'e-books'. Although Braille output of such texts is commonplace, the majority of blind people do not read Braille because it is difficult to learn & and the fine dots are not easy to distinguish by people with impaired tactile senses. Both of these are particular problems for people who become blind through old age, which is increasingly the principle cause of blindness. Moon could well be more suited to them and is the second most popular tactile script in the UK. However, its uptake has been severely hindered by the prior lack of a computer display method which limited readers to relatively few preprinted texts which were bulky (even compared to Braille books) and were very expensive to deliver compared to digital text files. This system might remove the obstacle to the development of Moon text, allowing it to grow in conventional as well as computer simulated usage.

As my force calculating algorithm for Moon text is really just a specific case of a general-purpose texture calculating algorithm, Moon text can be applied to any simulated objects, not just flat planes, just like other textures can. Therefore it can go beyond labelling simulated traditional tactile diagrams and also label full 3-dimensional haptic models.

The simulated Moon characters are a computer font so it is trivial to instantly customise their size & other parameters. This not only makes it highly adaptable to people with different tactile acuities, but also makes it suitable as training aid for reading paper Moon text. The trainee can start with large fonts and progress down to standard size. They could, if desired, be moved along the letters by an added helping push from the display. The customisability can be extended to a complete font replacement for tactile scripts other than Moon. It could even enable tactile computer output of ideographic scripts, including Japanese Kanji & traditional Chinese. Braille is commonly used for Chinese computer output but much is lost, particularly in poetic works, when the language is reduced to a particular phonetic transcription to fit the small available character set.

Many further features can be added. For example: shallow groove links between the characters to make it traceable like cursive text, raised line-rules to deter wandering off text lines and superimposing a paper texture to give a extra clue as to distances traversed. The latter was implemented by taking one of the original rough textures and summing it with the text texture.

7. CONCLUSION

Moon, and similar, tactile scripts can be dynamically displayed from computers by these means and applied to simulated tactile diagrams.

8. ACKNOWLEDGEMENTS

This work was funded by BTexact Long Term Research as part of the Multisensory Communications project, headed by Stephen Furner, and was filed as European Patent 01307951.2. I also wish to thank Prof. Helen Petrie of the National

Centre for Tactile Diagrams for suggesting ‘vBraille’ which lead to this & Dr. P.-L. Chau of Dept. Biochemistry, Cambridge University, for suggesting the extension to Chinese.

9. REFERENCES

H.Petrie, 1999. Personal communication.

RNIB, 2002. About Braille and Moon, http://www.rnib.org.uk/braille/welcome.htm. (Checked 2002/9/30.)

A.Hardwick, S.Furner, J.Rush, 1998. Tactile Display of Virtual Reality from the World Wide Web - a Potential Access Method for Blind People, IEE Displays 18 pp. 153-61.