How to Turn the Page

Yi-Chun Chu,[*] Ian H. Witten[*], Richard Lobb[†] and David Bainbridge[*]

[*]*Department of Computer Science*
University of Waikato
Hamilton, New Zealand
{ycc1, ihw, davidb}@cs.waikato.ac.nz

[†]*Department of Computer Science*
University of Auckland
Auckland, New Zealand
richard@cs.auckland.ac.nz

Abstract

Can digital libraries provide a reading experience that more closely resembles a real book than a scrolled or paginated electronic display? This paper describes a prototype page-turning system that realistically animates full three-dimensional page-turns. The dynamic behavior is generated by a mass-spring model defined on a rectangular grid of particles. The prototype takes a PDF or E-book file, renders it into a sequence of PNG images representing individual pages, and animates the page-turns under user control. The simulation behaves fairly naturally, although more computer graphics work is required to perfect it.

1. Introduction

Digital libraries invariably present their documents in a manner that is rather bland. Most collections show electronic text, frequently formatted for the screen in a way that is crude compared with typeset book pages. Designers of electronic books [1] pay more attention to the look and feel of the pages, with crisp text, clearly formatted and attractively laid out. Many digital library collections offer page images rather than electronic text, and although these are occasionally rather beautiful they are presented in a flat, two-dimensional manner.

Bibliophiles love books as much for the statements they make as objects as for the statements they contain as text. Indeed, early books were true works of art—beautifully-calligraphed poems inscribed into monumental stone steles in China; ancient scriptures etched onto palm leaves bound with string threaded through holes in India; and Western art treasures such as the *Book of Kells*, produced by Irish monks at the scriptorium of Iona 1200 years ago.

Beautiful books have always been prized for their splendid illustrations, for colored impressions, for decorated illuminated letters, for being printed on special paper, or uncommon materials, for unusual bindings, for their rarity and historic significance. And beauty is functional: these books give their readers an experience that is richer, more enlightening, more memorable, than the prosaic, utilitarian—often plain ugly—web pages offered by today's digital libraries.

The British National Library's "Turning the pages" project (see www.bl.uk/information/ttp.html) is a rare attempt to provide a reading experience that more closely resembles a real book. Readers sit at a touch screen showing a double-page spread of what appears to be a physical rather than an electronic book. Users wipe a finger across a touch-sensitive screen to metaphorically pick up a page and turn it. Pages look three-dimensional; the book's binding moves slightly in sympathy as a page is turned; page edges to right and left indicate how far through the book you are.

To accomplish this, photographs have been taken at (about ten) intermediate points during each page turn—so that what is displayed is a stored photo, not an artifact computed from a model of the book. There are many images—for example, one version of the system consumes 304 Mb for only twenty book pages (this includes zoomed-in versions of each page, and accompanying audio). The system is constructed using Macromedia Director.

The simulation is compelling, and users rapidly become absorbed in the book itself, turning pages unthinkingly—in Coleridge's words, they "willingly suspend disbelief" in these "shadows of imagination."[1] But the main drawback is that a slow animation of every page-turn must be painstakingly photographed in advance, for each book. This note describes an extension of the idea that dynamically texture-maps the current page's content onto a full three-dimensional model of the page-turn, making it easy to turn the pages of any book served up by a digital library.

[1] In his Biographie Literaria Samuel Taylor Coleridge (1772-1834) hoped his work would exhibit "a semblence of truth sufficient to procure for these shadows of imagination that willing suspension of disbelief for the moment, [...] which constitutes poetic faith."

0-7695-1939-3/03 $17.00 © 2003 IEEE

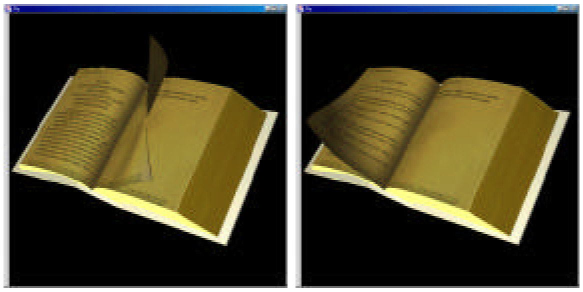

Figure 1 Java/OpenGL page turning prototype

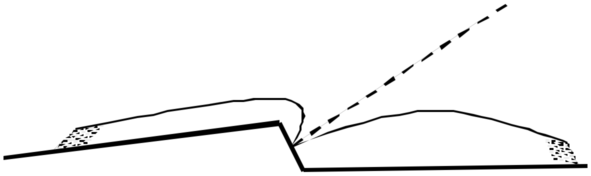

Figure 2 An open book

2. Modeling page-turns

We generate the dynamic behavior of individual page-turns using a model based on a mass-spring structure defined on a rectangular grid of particles. These are connected with primary mesh springs, diagonal bracing, and four-way flexural rigidity springs. Many of the springs are very stiff, leading to a stiff system of differential equations. The shape or "pose" of the page is computed as it is turned by a virtual hand, which holds the page at a given point and moves along a specified trajectory.

A Runge-Kutta method is used to solve for the dynamic behaviour of the page after each small movement of the virtual hand. The largish mesh (18 × 16) takes some time to settle, and the stiffness of the system necessitates small time steps. Consequently, the current implementation takes from minutes to hours for a single page-turn, precluding real-time operation. Though smaller less-stiff meshes can operate in near real time, their appearance is significantly less realistic. It may be possible to achieve a plausible simulation in real time with a faster differential equation solver, smaller mesh, and careful tuning.

For reasons of efficiency we decided to pre-compute a page-turn and save it on disk. Internally the mesh is a series of strips ultimately rendered as triangles. Each page turn comprises a sequence of frames, stored as mesh files. At run-time, these files are read in during the initialization stage. Our prototype implementation is written in Java and uses OpenGL to perform the rendering through the *GL4Java* API. OpenGL is rather quick at putting out textured triangle meshes, provided there is plenty of texture memory.

Different page-turns can be generated by manipulating parameters of the dynamic simulation. For example, one can set the virtual hand to grasp the page anywhere along its edge, alter the angle of its thumb and forefinger, and change its trajectory.

Numerous parameters control the properties of the paper itself. Pre-computing the turn removes the opportunity to vary these parameters, but risks giving an overly mechanical perceptual experience by repeating precisely the same movement. It would be possible to pre-compute several different variants, but this requires a significant amount of storage space. As a compromise, we achieve variety by pre-computing three different page-turns and interpolating between them in different, randomly-chosen, proportions.

3. Modeling the book

The pre-computed page-turn is superimposed on a model of the book's spine and covers that takes account of the page's position in the book. Figure 2 shows a cross-section. The dashed line represents an idealized version of the page in question partway through the page-turn. Its initial and final rest positions are shown: they are basically the same as the following and preceding pages' positions, respectively. The geometric configurations of these pages depend on how far through the book we are—and this also affects the angle of the spine.

Before the turn begins, the page in question rests on the book's next (right-hand) page. The influence of this initial position decays as the turn progresses until it dies out completely at what we call the "midpoint," whereupon the resting position of the previous (left-hand) page begins to have a gradually increasing influence, until at last the page rests on the preceding (left-hand) one. Throughout the turn, the page twists and bends according to the precomputed page model discussed in the preceding section.

The configuration of the cover is calculated from the thickness of the book relative to the page width, and the distance of the current page through the book. Given this information, the positions of the previous and next pages are modeled as spline functions whose parameters are determined by simple heuristics.

4. Putting it together

To operationalize a page-turn, the page is represented in terms of its divergence from the idealized flat-page model whose projection is shown dashed in Figure 2. The rotation angle increases from 0 to 180 degrees as the turn progresses. The page is a mesh, each mesh point being expressed as an offset from its corresponding position on the flat-page model. The value of the offset is formed by weighting and summing three things: (1) the effect of the initial position, whose weight decays as the turn proceeds until it disappears at the midpoint, (2) the effect of the final position, which gradually increases from the midpoint onwards, and (3) the effect of the pre-computed page-turn, which is a function of the angle of the idealized flat-page position.

Many practical problems arise. One is the need to respect the obvious physical constraint that a page may not penetrate its predecessor or successor. We accomplish this by performing the calculation for each mesh point in a radial coordinate system whose origin is the corresponding mesh position on the flat-page model and whose principal axis is normal to that surface. Then, interpenetration can be avoided by performing a min or max operation on each calculated coordinate position with respect to the corresponding mesh point on the page's predecessor and successor. Another problem is to determine the above-mentioned "midpoint" at which the page's initial position ceases to have any influence at all and its final position begins to take effect. Another is to divide the splines corresponding to the initial and final positions accurately into the appropriate number of equal parts. Finally there is a basic difficulty in mapping the 0 to 180 degree pre-computed page-turn into the more restricted angle through which the page must actually turn. We are continuing to investigate these problems. The reader's "willing suspension of disbelief" in the page's ethereal nature immediately breaks down with any distortion of the paper as it turns, and it is surprisingly difficult to preserve the model's integrity.

5. Conclusions

The page-turner exists in prototype form. Pages behave fairly naturally, although more work is required to perfect the computer graphics simulation—for example, saddle-shaped curvature gives the impression of a cloth-like material that is more flexible than paper. The prototype takes a PDF or E-book file, renders it into a sequence of PNG images representing individual pages, and animates the page-turns under user control. Ultimately we plan to integrate this functionality into the Greenstone digital library software [2].

As most users of Apple's System X will testify, well-designed, smoothly-animated screen effects (such as the "genie" put-away animation), in conjunction with touch-pad control, make computer-based activity a more relaxed, richer, higher quality experience than we are accustomed to. The British Library's work brings the same kind of quality to certain prized books. This project explores a generalization that will allow the contents of any digital library to be perused realistically.

Acknowledgements

We are grateful to Brian Wyvill of Calgary University for help and advice in the graphics area.

References

[1] Henke, H. (2001) *Electronic books and ePublishing: a practical guide for authors* Springer Verlag, New York.

[2] Witten, I.H. and Bainbridge, D. (2003) *How to build a digital library.* Morgan Kaufmann, San Francisco.