

A framework for interacting with paper

Peter Robinson, Dan Sheppard, Richard Watts, Robert Harding and Steve Lay

University of Cambridge, Computer Laboratory, New Museums Site, Pembroke Street, Cambridge, England CB2 3QG.

Abstract

This paper reports on ways of using digitised video from television cameras in user interfaces for computer systems. The DigitalDesk is built around an ordinary physical desk and can be used as such, but it has extra capabilities. A video camera mounted above the desk, pointing down at the work surface, is used to detect where the user is pointing and to read documents that are placed on the desk. A computer-driven projector is also mounted above the desk, allowing the system to project electronic objects onto the work surface and onto real paper documents. The animated paper documents project is considering particular applications of the technology in electronic publishing. The goal is to combine electronic and printed documents to give a richer presentation than that afforded by either separate medium.

This paper describes the framework that has been developed to assist with the preparation and presentation of these mixed-media documents. The central component is a registry that associates physical locations on pieces of paper with actions. This is surrounded by a number of adaptors that assist with the creation of new documents either from scratch or by translating from conventional hypermedia, and also allow the documents to be edited. Finally the DigitalDesk itself identifies pieces of paper and animates them with the actions described in the registry.

1. Background

In the 1970s, scientists at the Xerox Corporation's Palo Alto Research Center developed the *desktop metaphor* which made computers easy to use by making them look and act like ordinary physical desks. Electronic documents could be manipulated like paper documents, but the computer added powerful new facilities. This led some to predict that the *paperless office* would dominate within a few years. The trouble is that people like paper. It's portable, tactile and easier to read than a screen; today computers are used to generate far more paper than they replace.

Recent developments in computer hardware are greatly reducing the cost of attaching television cameras to computers. They have moved from being an expensive peripheral for specialists to a level comparable today with a monitor, and developments in technology will soon make the cost similar to that of a mouse. This raises the question of what new techniques will be appropriate when every computer routinely includes video input, possibly from several cameras.

Over the past few years, the University of Cambridge Computer Laboratory and the Rank Xerox Research Centre in Cambridge (formerly EuroPARC) have collaborated on re-

search into the use of video in user interfaces^{15, 18, 23}. This is not *virtual reality* where the user is immersed in a totally synthetic, computer-generated environment, often donning a special headset and even clothes; this is *augmented reality* where the computers operate through everyday objects in the real world. Such a system requires the computer to monitor activities and to deliver its contribution as unobtrusively as possible, suggesting the use of video and, to a lesser extent, sound for input and output. Of course, this merely reflects normal practice. We are used to pointing to interesting parts of documents and commenting on them; electronic enhancements should extend ordinary documents seamlessly.

The second motivation for this work is the development of electronic, multi-media publishing as an alternative to conventional publishing on paper. CD-ROM and on-line versions of reference books and fiction can enhance their conventional counterparts in a number of ways:

- They offer elaborate indexing, glossaries and cross-referencing.
- They allow non-linear progression through the text.
- Sound and moving images can be added.
- Sections can be copied into new documents.

However, screen-based documents have a number of disadvantages:

- People find screens harder to read than paper.
- Electronic bookmarks are less convenient than bits of paper.
- Adding personal notes to electronic documents is difficult.
- Writing, editing and proof-reading a non-linear, multi-media document is still a specialised and difficult task.

We are investigating ways of resolving these difficulties by publishing material as ordinary, printed documents that can be read in the normal way, enjoying the usual benefits of readability, accessibility and portability. However, when observed by a camera connected to a computer, they acquire the properties of electronic documents, blurring the distinction between the two modes of operation.

Our initial experiments have applied this technology to computer-assisted learning¹⁰. Earlier work with Computer Interactive Texts⁹ supplemented printed books with software that was an integral part of the educational package but which had to be run separately. The two parts can now be united. The first demonstration animates part of an elementary textbook on mathematics; applications to teaching English and digital electronics are currently being investigated. A separate paper discusses the user interface for preparing and presenting mixed-media documents¹⁶. This paper presents the internal architecture of the system. The early prototype systems to investigate video user interfaces^{7, 8, 12, 14, 17, 19, 20, 21, 22} used a fairly *ad hoc* approach. The whole system has now been re-engineered for more general use. In particular, questions of scale and of generality have been addressed. The resulting framework is now being used for further experiments on new applications of the technology.

2. System architecture

Moving from initial 'proof of concept' experiments to a full system has involved the design and implementation of a completely new framework for interacting with paper.

The first problem is to associate interactions with locations on printed documents. It would be attractive to embed the actual code for the interaction in the document, but the volume of data involved makes this impossible. The alternative is to mark the paper with a reference to the interactor and provide a central directory to return the code required over a network.

However, even these references are quite large. Several techniques are available for encoding digital information in printed material. Bar codes and special fonts for optical character recognition⁴ are in everyday use to identify things like products in supermarkets and cheques. More recently glyph codes¹¹ have been proposed as an alternative that can be printed relatively unobtrusively as part of the background of a document. This latter approach seemed particularly attractive as a way of simultaneously marking an active area of a document and identifying the associated interactor.

Hecht suggests that glyph codes should give an information density occupying only $0.3\text{mm}^2/\text{bit}$ with a 300 dpi printer and reasonable error correction¹¹. Our experiments⁵ confirmed that $1.6\text{mm}^2/\text{bit}$ were required for reasonable reliability if the documents were only printed at fax resolution. However, this does not work when using a high resolution camera over the desk because of difficulties resolving fine detail after applying the thresholding algorithm needed to handle uneven illumination. This means that even the interactor can not be identified in the space available and so a further level of indirection is needed. The solution is to use the identity of the document and the co-ordinates of the interactor on the page as an index to identify the interaction required. This information can be stored in the same central directory.

This leaves the problem of identifying the document. Newman suggests calculating a signature based on the arrangement of paragraphs, lines and words on the page¹⁴ but the information content is not sufficient for large scale use. Therefore, we mark the pages explicitly with unique identifiers. These encode the location of the directory as a network IP address and an index for the document within that directory. Again, glyph codes look attractive but proved unreliable. Bar codes require about $15\text{mm}^2/\text{bit}$ and OCR printing about $5\text{mm}^2/\text{bit}$, so the latter was adopted. Marking the paper documents with a unique identifier printed in an OCR font has the advantage that it can be transcribed manually to invoke interactions when the full digital desk is not available.

The registry providing these directory services thus lies at the centre of the system, acting as a long-term repository for representations of printed documents and the interaction objects associated with them. Information is added to the registry by direct editing or via an adaptor importing from some other form of hypertext. A document can be printed to paper via another adaptor and then animated by placing it on a DigitalDesk. The camera identifies the document and follows the pointer, and the associated actions are identified in the registry and invoked as appropriate with the results being projected back onto the paper. The overall framework is shown in Figure 1.

The remainder of this paper discusses these components in more detail. These are all written in Modula-3¹³, a high-level systems programming language that particularly lends itself to operation in a distributed environment. Finally, a typical application is described and some details of the implementation are given.

3. The Registry

The registry provides the central directory service for animated paper documents. It stores the image of each active document and the code of any interactors, together with cross references between these and indexes to identify them.

The main component is a *pseudo-window leaf* which corresponds to a unit of interaction on the DigitalDesk, not un-

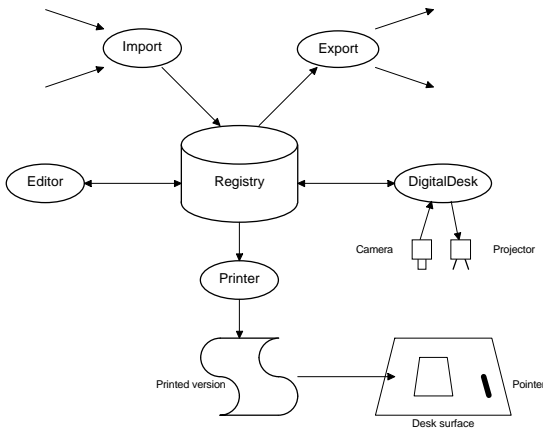


Figure 1: *The Animated paper document framework*

like a form in a conventional window system. These operate in two ways:

- as *primaries* – the master versions of the leaves which usually are editable but need not have been printed on paper, and
- as *surrogates* – which provide the co-operation with a printed page.

They can also operate as *factories* to reproduce the leaf, cloning its activities when the corresponding part of a printed page is copied and pasted.

Each leaf includes rendering information for display or printing. When a primary leaf is printed a *page representation* and a surrogate leaf are created in the registry. The page representation acts as an index identifying the interactor corresponding to particular co-ordinates on a page. It also allows further copies of the document to be printed with exactly the same interactive properties. Each interactor is associated with a particular piece of rendering information which refers back to the surrogate leaf which includes fragments of executable code.

The original intention was to store these as Modula-3 network objects³ which could be invoked remotely. Unfortunately, the Modula-3 type system requires the signatures of these remote methods to be known when the code is compiled, which makes dynamic operation difficult, so the code is stored in a single address space. However, this is just a temporary measure. A better long term solution would be to store complete programs as Java applets¹ or Obliq oblets⁶ which are more amenable to dynamic loading for remote execution.

The registry is made persistent through a hierarchical storage manager which saves the entire data structure as a series of files.

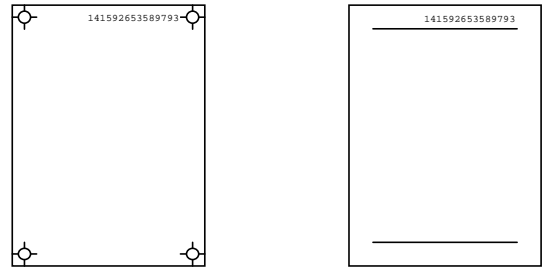


Figure 2: *Alternative layouts for animated pages*

4. Adaptors

The registry is accessed via a set of adaptors that allow the database to be built and edited, imported and exported to other forms of hypertext, and for documents to be printed for animation on a DigitalDesk.

4.1. Creation

Animated documents are created with a fairly conventional WYSIWYG editor. Text and diagrams are entered and amended in the usual way, building up a collection of rendering information in primary pseudo-window leaves. However, it is also possible to mark areas of the document as hyperlinks and to associate interactors with them. This adds code references to the rendering information.

One version of the editor actually operates on the DigitalDesk, which means that text, diagrams and interactors from other printed documents can be copied into the new document. If the other printed documents are active documents known to the system, this copying is entirely digital, just as it would be in a conventional word processor. However, text and pictures can also be copied from conventional printed documents by using the overhead camera to capture an image and passing any text through an optical character recognition system.

4.2. Printing

Another adaptor prints out documents from the registry onto paper so that they can be used for direct interaction on the DigitalDesk. The printed documents are annotated with marks in their corners to facilitate recognition and location on the desk top, and also have their unique identifier printed in an OCR font; see Figure 2.

Once the document has been printed, its page representation is retained in the registry as an immutable copy of its structure for future interaction. This allows the paper to continue working in the same way even if its electronic original is edited.

4.3. The DigitalDesk

The DigitalDesk actually animates the paper documents. This involves recognising that a page printed by the printing adaptor has appeared on the desk, determining its position, reading its unique identifier and locating any interactors. These are then marked by projected highlights. When these are activated by pointing, the associated code is recovered and executed, and the results projected back onto the paper.

The sequence of events is as follows:

- The characteristic corner marks or rules trigger the system's detection of an active page on the desk.
- The unique identifier is read using OCR.
- This encodes both the address of the originating registry and the page representation within that registry.
- The rendering information associated with the page representation specifies the active areas by the co-ordinates of their pseudo-window leaves on the page.
- Pointer events in the active areas are passed to the surrogate leaves which invoke the associated code.
- Results are projected back through other leaves.

We are currently working on improved page recognition techniques that will remove the requirement for characteristic registration marks and printed identifiers.

4.4. Import and export

Other forms of hypertext can be absorbed into the animated paper document system. For example, paper access to the World Wide Web² is possible through an adaptor. Given a uniform resource locator (URL), this captures the information on the associated web page in the registry. The page can then be printed on paper and links activated by placing the paper on a DigitalDesk and pointing. The page's identifier and the co-ordinates of the link are looked up in the registry to yield the appropriate activity and the results projected back onto the desk.

The interactions afforded by animated paper are considerably richer than straightforward HTML but if a document is sufficiently simple, it can be exported as HTML.

5. Direct interaction

Figure 3 shows a photograph of the current DigitalDesk. The camera and projector are mounted above the desk to give an A3-sized working area.

The system acts rather like a colleague or instructor. The user places a printed page on the desk, roughly the right way up and clearly visible, to 'show' it to the system in order to interact with it.

When the page has been located and recognised, a transformation is set up between the pseudo-window leaves attached to its page representation and physical co-ordinates



Figure 3: *The DigitalDesk in use*

on the desk top. The leaves thus become part of the projected window system.

A pen with a light-emitting diode in its tip is used for pointing. This is recognised by the camera system and converted to co-ordinates using a transformation calculated by occasional registration. It would be possible to use a conventional graphics tablet, but the light pen has the advantage that it works perfectly well over a stack of paper on the desk. The events are passed back through the window system to the leaves.

The paper itself can be annotated with an ordinary pen and the image captured and passed back to an application either as an image or as text derived by conventional optical character recognition.

6. Applications

The animated paper documents project involves work both on the underlying technology described here and also on new applications to prepare and present mixed-media publications.

One of the first applications is a system to assist with teaching mathematics¹⁰. Figure 4 shows an animated page from a textbook which presents a tutorial on the graphs of

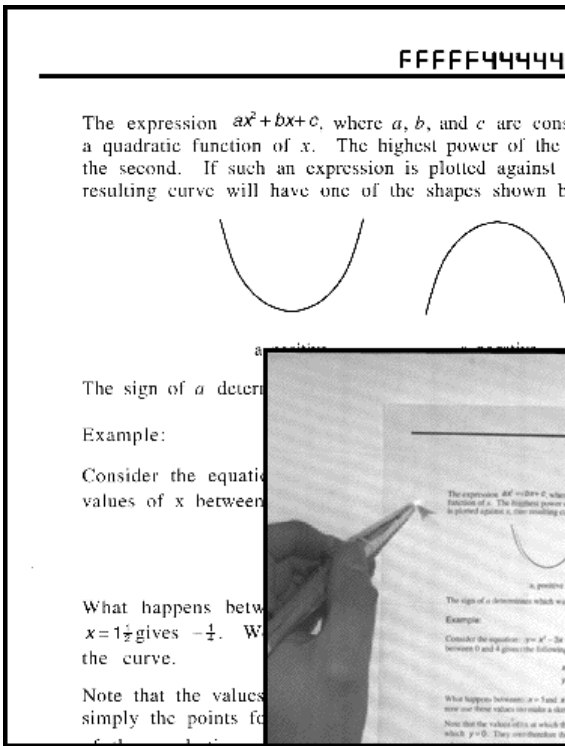


Figure 4: A page of an animated maths book with the camera's view inset

quadratic equations. The student can experiment with the book by writing coefficients into a general equation on the page and seeing the resulting graph drawn into a space on the page. Alternatively the system can project the coefficients and the student draw the graph in the space; this is then captured by the system and assessed.

Several further applications are being developed using this framework:

- An animated version of a story book will allow children to explore grammatical structures in the text.
- A computer-aided design system will allow finite state machines to be drawn and the corresponding hardware implementation displayed.
- Import and export adaptors have been written to allow interaction with printed version of World Wide Web pages.

7. Implementation

The current system is mostly built from commodity hardware. The registry runs on a Sun workstation under the Solaris operating system and the DigitalDesk runs on a PC under Linux. However, the software is all written in Modula-3 and so is portable to other platforms as appropriate. The DigitalDesk display is provided by a Proxima 1024x768 colour

LCD panel attached to a high power overhead projector and its camera is a JVC TK-F7300. This uses a prismatic arrangement to sub-sample a 756x576 image by a factor of 2, 3 or 6 in each direction to give a maximum resolution of 4536x3456. When mapped onto an A3 working surface, this gives a resolution of about 10 pixels/mm which is sufficient for OCR.

The registry and adaptors are written to work as a distributed system and documents can be printed, the paper taken to a geographically remote site, placed on a DigitalDesk there and animated through communications with the home registry that is identified by the unique identifier printed on the document. However, it is important that the system should also be able to operate when free-standing. This is achieved by taking a copy of the registry and the relevant adaptors and 'publishing' them on a CD-ROM. This can then accompany a printed version of a book and be used to animate the book on any computer workstation equipped with appropriate DigitalDesk hardware.

8. Conclusions

We have developed the underlying technology of the DigitalDesk and used it as a vehicle to demonstrate mixed-media documents adding electronic annotations to printed pages. This has prepared the way for widespread use of animated paper documents as the relevant hardware becomes more generally available.

In this paper we have described a framework for animated paper documents that provides an environment for further experiments with new applications and new user interfaces. This provides a central registry for document information, a set of adaptors to edit, print and interact with animated paper documents and further adaptors to import and export other forms of hypertext.

The first animated paper documents have already been tested within the framework and further mixed-media publications are being prepared. We are continuing to work on the underlying technology and on taking the applications forward into educational material for other disciplines and into other areas of publishing.

Acknowledgements

The original DigitalDesk was built by Pierre Wellner, a research student in the Computer Laboratory at the University of Cambridge sponsored by Rank Xerox. Current work on animated paper documents is sponsored by the EPSRC under grant GR/J65969.

References

1. Ken Arnold & James Gosling: *The Java programming language*, Addison-Wesley, 1996, ISBN 0-201-63455-4.

2. Tim Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen & Arthur Secret: *The World-Wide Web*, Communications of the ACM 37(8), August 1994, pp 76-82.
3. Andrew Birrell, Greg Nelson, Susan Owicki & Edward Wobber: *Network objects*, Proceedings 14th ACM Symposium on Operating System Principles, December 1993, pp 217-230.
4. Peter Brophy: *Computers can read*, Gower Publishing Company, 1986, ISBN 0-291-39701-8.
5. Lewis Brown: *Implementation of software for an encrypting photocopier*, CST dissertation, University of Cambridge, June 1995.
6. Marc Brown & Marc Najork: *Distributed active objects*, DEC SRC Research Report 141a, April 1996; published in Computer Networks and ISDN Systems 28, 1996, pp 1037-1052.
7. Kathy Carter: *Computer aided design – back to the drawing board*, Proceedings of the International Symposium on Creativity and Cognition, Loughborough, April 1993.
8. Steve Freeman: *An architecture for distributed user interfaces*, Technical Report 342 (PhD dissertation), University of Cambridge Computer Laboratory, July 1994
9. Robert Harding: *Computer interactive texts*, Proceedings fourth annual conference on technology in collegiate mathematics, Addison-Wesley 1993, pp 132-138.
10. Robert Harding, Steve Lay, Peter Robinson & Richard Watts: *New technology for interactive computer aided learning*, Journal of the Association for Learning Technology 5(1), 1997, pp 6-12.
11. David Hecht: *Embedded data glyph technology for hardcopy digital documents*, Proceedings of the SPIE Color Hard Copy and Graphic Arts Conference, February 1994.
12. Wendy Mackay, Gilles Velay, Kathy Carter, Chaoying Ma & Daniele Pagani: *Augmenting reality – adding computational dimensions to paper*, Communications of the ACM 36(7), July 1993.
13. Greg Nelson (editor): *Systems programming with Modula-3*, Prentice-Hall, 1991.
14. William Newman & Pierre Wellner: *A desk that supports computer-based interaction with paper documents*, Proceedings of the ACM Conference on Human Factors in Computing Systems, Monterey, May 1992.
15. Peter Robinson: *Virtual offices*, Proceedings of Royal Society discussion meeting on Virtual reality in society, science and engineering, BT Publication SRD/R5/1, July 1995.
16. Peter Robinson, Dan Sheppard, Richard Watts, Robert Harding & Steve Lay: *Animated paper documents*, Proceedings 7th International Conference on human-computer interaction, HCI '97, San Francisco, August 1997.
17. *The EuroCODE project*, Rank Xerox Research Centre Cambridge, June 1993.
18. Quentin Stafford-Fraser: *Video augmented environments*, PhD dissertation, University of Cambridge Computer Laboratory, March 1996.
19. Quentin Stafford-Fraser & Peter Robinson: *Bright-Board - a video augmented environment*, Proceedings of the ACM Conference on Human Factors in Computing Systems, Vancouver, April 1996.
20. Pierre Wellner: *The DigitalDesk calculator – tangible manipulation on a desk-top display*, Proceedings of the ACM Symposium on User Interface Software and Technology, Hilton Head, November 1991.
21. Pierre Wellner: *Tactile manipulation on the DigitalDesk*, Video presentation at the ACM Conference on Human Factors in Computing Systems, Monterey, May 1992.
22. Pierre Wellner: *Interacting with paper on the DigitalDesk*, Communications of the ACM 36(7), July 1993.
23. Pierre Wellner: *Interacting with paper on the DigitalDesk*, Technical Report 330 (PhD dissertation), University of Cambridge Computer Laboratory, March 1994.