DML 2008

Michael Doob

Small Scale Retrodigitization

In: Petr Sojka (ed.): Towards Digital Mathematics Library. Birmingham, United Kingdom, July 27th, 2008. Masaryk University, Brno, 2008. pp. 103--113.

Persistent URL: http://dml.cz/dmlcz/702534

Small Scale Retrodigitization

Michael Doob

Department of Mathematics, The University of Manitoba, Canada
E-mail: `mdoob@ccu.umanitoba.ca`

Abstract. The digitization of papers born in the print-only era is vital for the health of the mathematical record. Many large scale retrodigitization projects are underway and, at this point, probably more that half of the mathematical history has been finished. Many smaller journals and books remain to be done. This paper gives a framework within which these may also be completed. It uses the digitization of the Canadian Journal of Mathematics (53,000 pages), completed as a one-man project over a few months, as the working example. The project described herein not only may be used as a model for similar efforts but also indicates some interesting problems yet to be solved.

Key words: home retrodigitization, tesseract, OCRopus, OCR, Canadian Journal of Mathematics

1 Introduction

It is a cliché to observe that during the course of the 90s the publication of mathematical research moved from the age of manuscripts to the age of TEX. The resulting documents, born digital, were in a malleable format which allowed for easy adaptation to the internet. Indeed, packages such as Sebastian Rahtz' hyperref [13] allowed for both internal and external hyperlinks, repositories such as arXiv [1] allowed easy access.

Mathematics is unusual in that its literature has staying power. Indeed, papers many decades old often recur in the bibliographies of recent papers [5]. For this reason the desire to access papers born in print via the internet is completely reasonable and expected. Indeed, many journals, some of which have a history of several hundred thousand pages, have been made available on the internet by retrodigitization. Many of these journals have been amalgamated into digital libraries, some of which are quite spectacular. The leading example is the NUMDAM collection in Grenoble, France [10].

Retrodigitization takes place in a sequence of steps:

1. The printed pages are scanned to get a graphic image. This is called the *graphics plane*.
2. A reader uses optical character recognition (OCR) to extract text and possibly other information from the graphic plane. The resulting file is called the *text plane*.

Petr Sojka (editor): DML 2008, Towards Digital Mathematics Library, pp. 103–113.
ISBN 978-80-210-4658-0

3. The metadata of the item being digitized are constructed.
4. Files suitable for the internet are created.
5. The created files are collected in a library.

The purpose of this paper is to concentrate not on the large projects that are under way but rather to focus on retrodigitization on a small scale. The requires differing approaches in carrying out the steps given above. For example, for larger projects it makes sense to have some of them outsourced, perhaps offshore, but for smaller ones this is simply impracticable. In particular, a specific project will be described: the retrodigitization of the Canadian Journal of Mathematics (CJM) for the Canadian Mathematical Society (CMS). A complete description will be given, starting with the scanning of the backfile (some 45,000 pages in over 4,000 articles) and concluding with the appearance of the files on the internet. Since this project was carried out by one person as a part-time effort over a couple of months, it is hoped that the experiences described herein will be helpful in promoting similar projects. It is also hoped that the experiences and problems releated will pique the interest of those who create the tools that enable others to do digitization projects more easily.

2 Scanning the Hard Copy: Construction of the Graphics Plane

The first choice that affects the nature of the scanning is whether it will be destructive or nondestructive. Generally speaking, it is much easier to work with destructive scanning, that is, where the pages of the journals or books are cut out of the bindings and handled as individual sheets; this allows the use of a sheet feeder during the scanning process. The pages may be removed using a scoring knife and a straight edge; an issue of a few hundred pages can be cut from the binding in three or four strokes. Even easier, a local photocopy centre will often have the equipment necessary to do the cutting at a minimal cost. In the CMS project, it was done at the University copy centre for 50¢ per issue (all prices in this paper are in Canadian dollars).

The next step is to choose the hardware and software to be used in the scanning process itself. To do this, it is first necessary to decide on the parameters of the scanning (resolution, colour depth and the like). Fortunately, the is a set on best-practice recommendations published by the International Mathematical Union [2] that can be used for this purpose. In general, for normal black and white text, the minimum resolution is 600 dots per inch, and there are many pieces of equipment that are both inexpensive and able to reach this minimum. In fact, many Departmental photocopiers are able to do scanning at this resolution, although it is not usually the default. Scanners with automatic sheet feeders are easily obtainable for $1,500.

The chosen hardware will come with some scanning software. Most commonly it expects to be hooked up to a computer running Windows XP. More generally, many printers are TWAIN compliant. This standard means that the scanner will run with similarly compliant software (most of which are Windows or Mac OS oriented). Perhaps it is worth noting that a computer with Windows XP licence can be purchased for about half the cost of the scanner.

Similarly, scanners which are SANE compliant can run on appropriate software (usually Linux oriented).

For the CMS project we bought a Hewlett-Packard Scanjet 8390.

Once the hardware and software are in place, the format of the scanned file must be decided. The most standard format at this time is TIFF (tagged image format file), a format that may be chosen to be lossless (losing no information upon compression). The files may be of single pages, or they may be multipaged. These files tend to be very big, and the Group 4 fax compression is often used for easier storage with no loss of data upon decompression. The compression of black and white text often reduces the file size by over 90%. The freely available ImageMagick [8] suite of tools were particularly useful for this purpose.

Whatever format is used, it essential that the images be checked very carefully. While some computer verification and correction (despeckling and alignment) is possible, ultimately visual inspection of each page in necessary. A significant amount of time (and, perhaps, money) must be allocated for this purpose. It is important to remember that, although the scanned files are not the final product, they are ultimately the most expensive to produce; it is crucial that these be made carefully and archived properly, for they are the authoritative record, and the derivative files (text, pdf, html) used by other applications ultimately depend on them. In the near future improvements in OCR software are expected. The recognition of mathematical expressions, for example, might be greatly improved, and if this happens, it will be relatively inexpensive to run this newer software on the (archived) TIFFs.

Here is how this process worked for the CMS project: the scanner could do two-sided scanning, and could handle 100 pages at a time. Scanning these pages at 600 dots per inch took nine minutes (an oddity for our scanner: the pitch of the sheet feeder is slightly higher when the last sheet goes in; the next hundred pages can then be dropped in for “continuous” feeding). Since our goal was 1,000 pages per day, the scanning phase was normally done in under two hours.

The scanned pages were stored as single-page TIFFs. This reflected the view that the page was the essential unit, and, depending on the application, articles, numbers or full volumes could be constructed as needed.

The second stage was to check the scanned images. Some of the problems encountered:

- Two pages would stick together resulting in two pages unscanned.
- The pinch rollers wouldn’t pull evenly resulting in skewed text.
- A little leftover glue on one edge of the page would cause the paper to slip and pass through the scanner at an inappropriate angle. This would result in a skewed image or a paper jam.
- A single sensor wouldn’t read and one pixel would be missing in each line producing a thin white vertical line (interestingly enough, this seems to

be minor enough to leave the OCR unaffected; nonetheless, it was really annoying visually).
- The corner of a page would fold over the text.
- The page would wrinkle inside of the scanner.
- The page would tear inside the scanner.

The obvious first check was to ensure that the number of pages matched the number of files. The next step was to put the images on the screen as a slide show. (I found that one image per second was enough to check for correct pagination and alignment.) Occasional rescans were necessary, but only of one page at a time since that was the unit of storage. Normally this second stage could be done in less than an hour. Unlike the first stage, it really required constant attention. Once the TIFF files were satisfactory, they were ready for the OCR software.

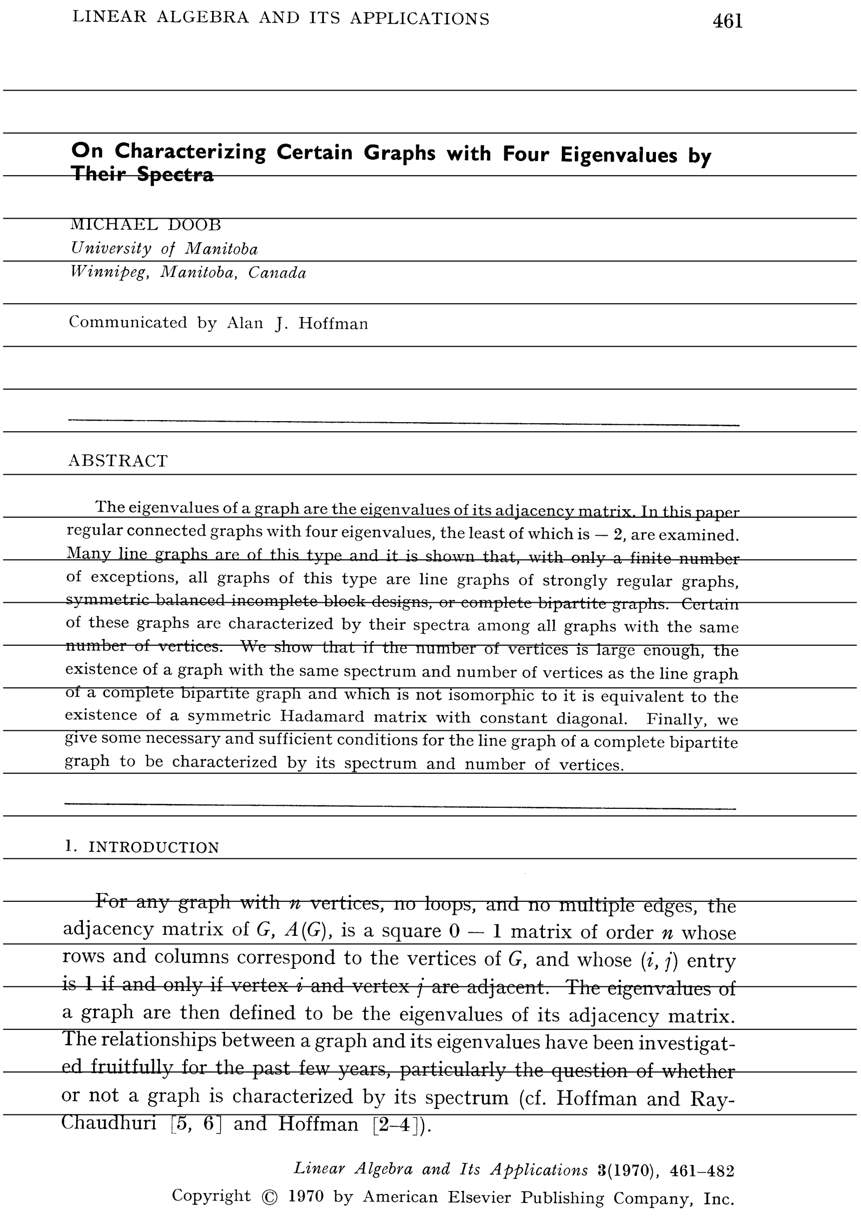
It is interesting to note the precision of the eye for these visual checks. In Figure 1 the lines are horizontal and the text has been rotated one degree clockwise. It is astonishingly easy to spot the difference in alignment. Since the recommended alignment [2] is within two degrees, only a glance is necessary to assure the compliance.

3 Optical Character Recognition: Constructing the Text Plane

The choice of OCR software can be tricky. It is in part determined by the text being scanned. Multilingual texts, perhaps with letters from non-Latin character sets, require more sophistication from the software. While there is some open-source software available, the application is still at an early stage, and commercial software may be more appropriate (but see Section 7). Most such software has a free "test drive" version that can be used for a trial period. It is a good idea to utilize such a version to see how it works with your scanned images and, in particular, with the physical characteristics of the material being scanned. For example, it is possible to get dramatically better results by manually adjusting the brightness and contrast settings used for the scanning. In addition, OCR software will usually come with a learning mode to get better recognition. To check the quality of the OCR software, it is absolutely essential that the software produce a viewable text file of the characters recognized from the images as they are processed.

For those more attuned to open source software, there is the tesseract OCR software, which was originally developed by Hewlett-Packard and is now released as open software [7]. Further, there is a Google project [6] which will allow tesseract to read uncompressed TIFF images and a special front end OCRopus [11] for textual analysis to be used under (some versions of) Linux.

Once the OCR software is chosen, the images may be processed, and files containing the characters recognized as well as the files for public viewing are produced. The most common form of the latter is text-searchable PDF files. Like the TIFFs, the PDF files should be visually checked also.



LINEAR ALGEBRA AND ITS APPLICATIONS 461

On Characterizing Certain Graphs with Four Eigenvalues by Their Spectra

MICHAEL DOOB
University of Manitoba
Winnipeg, Manitoba, Canada

Communicated by Alan J. Hoffman

ABSTRACT

The eigenvalues of a graph are the eigenvalues of its adjacency matrix. In this paper regular connected graphs with four eigenvalues, the least of which is -2, are examined. Many line graphs are of this type and it is shown that, with only a finite number of exceptions, all graphs of this type are line graphs of strongly regular graphs, symmetric balanced incomplete block designs, or complete bipartite graphs. Certain of these graphs are characterized by their spectra among all graphs with the same number of vertices. We show that if the number of vertices is large enough, the existence of a graph with the same spectrum and number of vertices as the line graph of a complete bipartite graph and which is not isomorphic to it is equivalent to the existence of a symmetric Hadamard matrix with constant diagonal. Finally, we give some necessary and sufficient conditions for the line graph of a complete bipartite graph to be characterized by its spectrum and number of vertices.

1. INTRODUCTION

For any graph with n vertices, no loops, and no multiple edges, the adjacency matrix of G, $A(G)$, is a square $0-1$ matrix of order n whose rows and columns correspond to the vertices of G, and whose (i, j) entry is 1 if and only if vertex i and vertex j are adjacent. The eigenvalues of a graph are then defined to be the eigenvalues of its adjacency matrix. The relationships between a graph and its eigenvalues have been investigated fruitfully for the past few years, particularly the question of whether or not a graph is characterized by its spectrum (cf. Hoffman and Ray-Chaudhuri [5, 6] and Hoffman [2–4]).

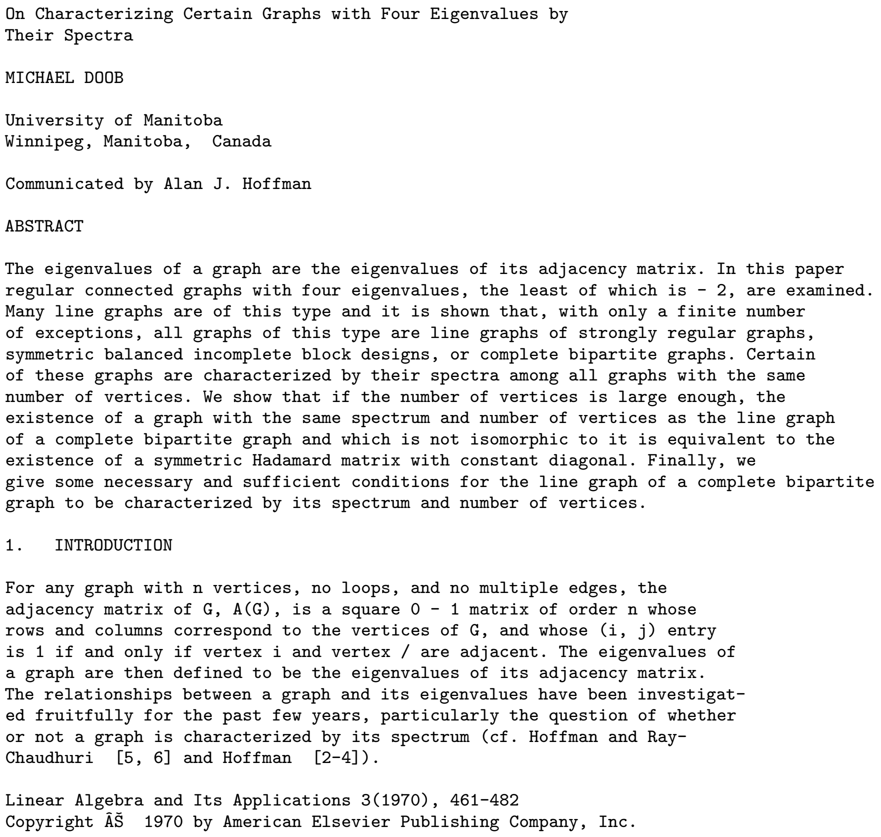
Linear Algebra and Its Applications 3(1970), 461–482
Copyright © 1970 by American Elsevier Publishing Company, Inc.

Fig. 1. Horizontal lines with text rotated by one degree

For the CMS project, the ABBYY FineReader software was chosen. This commercial software ($400 for the single-user Windows edition) easily met our multilingual requirements (mostly English and French, with a little German) and fulfilled the requirements described above. Each 1,000 page batch would take four to five hours to process, and the jobs were typically run overnight. This gave us the text output from the scan, a compressed copy of the scanned image, an image-only pdf file and a text-searchable pdf file. These were visually checked by printing the PDF files with 4-up duplex printing. Sometimes errors would show up even at this stage, but it was pretty rare.

Finally, it is necessary to gather pages into files for each document. If PDF files are being used, there is freely available software such as pdftk [12] that will do this easily. It is helpful to have the file names reflect the content (journal, year, volume, and page range).

Another popular lossless format in the mathematical world is DjVu, which can give even higher compression ratios.



```
On Characterizing Certain Graphs with Four Eigenvalues by
Their Spectra

MICHAEL DOOB

University of Manitoba
Winnipeg, Manitoba,  Canada

Communicated by Alan J. Hoffman

ABSTRACT

The eigenvalues of a graph are the eigenvalues of its adjacency matrix. In this paper
regular connected graphs with four eigenvalues, the least of which is - 2, are examined.
Many line graphs are of this type and it is shown that, with only a finite number
of exceptions, all graphs of this type are line graphs of strongly regular graphs,
symmetric balanced incomplete block designs, or complete bipartite graphs. Certain
of these graphs are characterized by their spectra among all graphs with the same
number of vertices. We show that if the number of vertices is large enough, the
existence of a graph with the same spectrum and number of vertices as the line graph
of a complete bipartite graph and which is not isomorphic to it is equivalent to the
existence of a symmetric Hadamard matrix with constant diagonal. Finally, we
give some necessary and sufficient conditions for the line graph of a complete bipartite
graph to be characterized by its spectrum and number of vertices.

1.   INTRODUCTION

For any graph with n vertices, no loops, and no multiple edges, the
adjacency matrix of G, A(G), is a square 0 - 1 matrix of order n whose
rows and columns correspond to the vertices of G, and whose (i, j) entry
is 1 if and only if vertex i and vertex / are adjacent. The eigenvalues of
a graph are then defined to be the eigenvalues of its adjacency matrix.
The relationships between a graph and its eigenvalues have been investigat-
ed fruitfully for the past few years, particularly the question of whether
or not a graph is characterized by its spectrum (cf. Hoffman and Ray-
Chaudhuri  [5, 6] and Hoffman  [2-4]).

Linear Algebra and Its Applications 3(1970), 461-482
Copyright ÂŠ  1970 by American Elsevier Publishing Company, Inc.
```

Fig. 2. One page of extracted text

4 Constructing the Metadata

The creation of metadata files should be given high priority in any retrodigitization project. These files will allow the efficient construction web pages and the interaction with other databases. They are the key to communicating the rest of the electronic world.

Most metadata files use some variant of XML, that is, they are text files with named fields starting with something like `<name>` and ending with something like `</name>`. They may be nested.

Here is a sample of metadata for an article (this particular format is used by Google):

```
<article>
  <front>
    <journal-meta>
      <journal-title>Linear Algebra and Its Applications</journal-title>
      <abbrev-journal-title>Linear Algebra and Appl.</abbrev-journal-title>
      <issn>0024-3795</issn>
      <publisher>
        <publisher-name>Elsevier</publisher-name>
      </publisher>
    </journal-meta>
    <article-meta>
      <title-group>
        <article-title>
        On characterizing certain graphs with four eigenvalues
        by their spectra
        </article-title>
      </title-group>
      <contrib-group>
        <contrib contrib-type="author">
          <name>
            <surname>Doob</surname>
            <given-names>Michael</given-names>
            <suffix></suffix>
          </name>
        </contrib>
      </contrib-group>
      <pub-date pub-type="pub">
        <year>1970</year>
      </pub-date>
      <volume>3</volume>
      <issue></issue>
      <fpage>461</fpage>
      <lpage>482</lpage>
    </article-meta>
  </front>
</article>
```

The names used and data included are more or less arbitrary. A suggested and widely-accepted minimum for different data types of publications is given for the mini-dml [9]. Articles, for example, should have metadata that include the author, title, journal, year, volume, number and page range.

The importance of the metadata files is clear. They are key to constructing other files, html ones, for example, that will allow access to the retrodigitized images.

5 Verification of the Metadata

Checking the validity of the metadata, while absolutely vital to any retrodigitization project, is difficult, even for small projects. The best hope is to have as many eyes as possible do the verification. These may be human eyes (not very good at the job), or computer eyes (better at the job with proper information and instruction). There are several possible ways of combining the process:

1. Inputing the data into a form by a human.
2. Using the text extracted by the OCR.
3. Using online databases.

The accuracy of the metadata can be enhanced by using the different possibilities to cross check the results.

The online databases using Math Reviews (MathSciNet) and Zentralblatt are particularly helpful. In particular:

Remark 1. Anything that can be checked by a computer program should be.

Also, as we see from Figure 1,

Remark 2. The eye is a precision instrument. Use it.

For example the following is the download of an article from MathSciNet in a text format:

```
%0 Journal Article
%A Doob, Michael
%T On characterizing certain graphs with four eigenvalues by their spectra
%J Linear Algebra and Appl.
%V 3
%D 1970
%P 461--482
%L MR0285432 (44 \#2650)
```

Relatively simple scripts can be used to transform the structure to an XML format if desired. Perhaps it is worth noting that the online databases are highly accurate but not error free.

For the CMS project we downloaded metadata from MathSciNet using the bibtex format.

```
\bib{MR0285432}{article}{
   author={Doob, Michael},
   title={On characterizing certain graphs with four eigenvalues by their
   spectra},
   journal={Linear Algebra and Appl.},
   volume={3},
   date={1970},
   pages={461--482},
   review={\MR{0285432 (44 \#2650)}},
}
```

The error rate for files in this format was about 1%. Most of the errors were pretty trivial (such as a missing period or space) but a very small number of "real" errors showed up. Relatively simple scripts (using Perl and AWK) were used to transform the structure to an XML format.

Even with this simple scheme exceptions arose. For example the MathSciNet returned the pages as a range, e.g., `461--482`. The metadata was tagged with `<fpage>` and `<lpage>`, requiring only a straightforward translation, *unless there was only one page*. Little surprises of this type were commonplace.

A glance at Figure 2 reveals all of the metadata sitting as text right on the page. This gives us a powerful method of cross-checking the metadata. Fortunately it was possible to use this technique for the verification of much of the CMS metadata. When the first page of the article did not have all the desired information, the annual index was used instead.

A significant but perhaps not initially apparent choice for the metadata is the character encoding used. For example, the final letter in the word "Poincaré" must be properly interpreted by the software using the metadata. For alphabetizing author names, as another example, it is crucial that accented initial letters, accented or otherwiae, be understood properly. The emerging standard at this time is UTF-8.

The character encoding had another somewhat unexpected application. Author names with accents are usually stored in databases using TeX format. For example, Poincaré may appear as `Poincar\'e` or `Poincar\'{e}` while Erdős may appear as `Erd\H{o}s`, `Erd\H os` or even `Erd{\H o}s`. In addition, our CMS internal database also used TeX for accented names. In such a situation, how can author names be verified? Our answer was to translate the names into the UTF-8 character encoding, and then just compare them as text strings.

This necessitates a script to translate the TeX encodings of accented characters used by Mathematical Reviews and our internal database to the UTF-8 encoding. This is a bit challenging since, as observed previously, there are many possible ways of encoding the same character in TeX, and indeed many of them turn out in practice.

Fortunately all UTF-8 characters have a (well-defined) name as part of the Unicode standard [15]. For example, the last letter in Poincaré has a name: LATIN SMALL LETTER E WITH ACUTE, and an encoding vector position of `00E9` and is displayed using the two bytes `C3` and `89`. Hence all that is necessary is to have a listing of the different ways that the letter is encoded in TeX: `{\'e}`,

`\'e` and `\'{e}` and then translate it to the appropriate UTF-8 code. Similarly the character LATIN SMALL LETTER C WITH CEDILLA has a UTF-8 character encoding vector position of `00E7`, is printed with the two bytes `C3` and `87`, and may be encoded in TEX by `{\c{c}}`, `\c{c}` or `\c[:blank:]c`.

The CMS database uses the author's own coding for the metadata. One might think that all author inputs could be covered with a few regular expressions, but this underestimates the creativity of authors. It turns out to be easier just to make an (extensible) list for each character. Once this list is set, the translation code is trivial (a much more extensive translation package may be found at [14]).

6 And in the End...

The goal is to have the files available on the web. Most of the work is done now. The final steps are routine:

1. Have the OCR software produce a text-searchable `pdf` file for each page.
2. Use the metadata to gather pages into individual articles.
3. Use the metadata to make a table of contents for each volume.
4. Use the metadata to make a global index.
5. Add an html wrapper to allow easy access to the `pdf` files.

So what were the actual costs for the CJM retrodigitization? As we have seen, there are several items that need to be considered. Here is some of our CMS results:

Table 1. Costs for retrodigitizing 53,000 pages (Canadian dollars)

Scanner	$1,700
OCR software	$ 400
Binding removal	$ 125
Computer	$2,800
Printer costs	$ 300
Personal time	∞

The total cost in dollars turned out to be quite modest: about $5,000 or somewhat under 10¢ per page. Well, modest at first blush, but what about the cost of labour so conveniently omitted? The project itself consumed about 300 hours of effort, all of which was "volunteer". This doesn't include the development time for the associated computer scripts, several dozen of which emerged; so there is obviously a significant unreported expense. On the other hand, for small scale retrodigitization projects, volunteers for routine tasks are often available, so these expenses may not have to be allocated directly.

To sum up, the experiment to see if the retrodigitization of a modest-sized journal could be accomplished by one person in a reasonable amount of time should be considered successful. The hardware and software that have appeared in the last two years bring such projects within the financial reach of smaller organizations. The resulting files are easy to distribute. In fact, the PDF files of the entire predigital history of the Canadian Journal of Mathematics, can fit on one USB memory stick.

7 Back to the Future

While retrodigitization projects of the type described in this paper are certainly feasible, there is much that could be done to enable such projects to be done more easily. In particular, having a full suite of freely available software would be helpful. As a first step, the CMS is making the 53,244 scanned images of the CJM (1949–1996) freely available as a testbed.

Here are some possible further directions:

- Use tesseract with the cms-cjm testbed.
- Put a front end on tesseract to improve its user interface.
- Create tools for adding hyperlinks to the bibliography, in particular, of scanned mathematical papers (see [4]).
- Build a package to automate the procedures described in this paper.

References

1. The home web site for ArXiv is `http://arxiv.org/` and is hosted by the Cornell University Library. The history of ArXiv is given in the article at `http://en.wikipedia.org/wiki/ArXiv`.
2. `http://www.ceic.math.ca/Publications/retro_bestpractices.pdf`.
3. Keith Dennis has had some encouraging results using Perl scripts developed by his working group at Cornell. His software has only been circulated informally.
4. Dennis, K., Michler, G. O., Schneider, G., Suzuki, M.: Automatic reference linking in distributed digital libraries, CVPRW 2003, Conference on Computer Vision and Pattern Recognition Workshop, paper #26, Volume 3 (Workshop on Document Image Analysis and Retrieval), 5 pp. (2003).
5. Ewing, J.: Measuring Journals. Notices of the AMS, 1049–1053, (2006).
6. The project location is `http://code.google.com/p/tesseract-ocr`.
7. Described at `http://en.wikipedia.org/wiki/Tesseract_(software)` and announced at `http://google-code-updates.blogspot.com/2006/08/announcing-tesseract-ocr.html`.
8. The main site is at `http://www.imagemagick.org/script/index.php`.
9. A full description of this project is at `http://minidml.mathdoc.fr/`.
10. NUMDAM (in English) is at `http://www.numdam.org/?lang=en`.
11. See `http://en.wikipedia.org/wiki/OCRopus`.
12. The home page for this software is `http://www.pdfhacks.com/pdftk/`.
13. Documentation for the hyperref package can be found both at `http://en.wikibooks.org/wiki/LaTeX/Packages/Hyperref` and at `http://www.tug.org/applications/hyperref/`.
14. `http://www-sop.inria.fr/apics/tralics` specifically translates LaTeX to XML.
15. `http://www.unicode.org/charts` contains a list of the standard character names.