Mathematical Knowledge Browser with Automatic Hyperlink Detection*

Koji Nakagawa and Masakazu Suzuki

Faculty of Mathematics, Kyushu University,
Kyushu Univ. 36, Fukuoka 812-8581, Japan
{nakagawa, suzuki}@math.kyushu-u.ac.jp

Abstract. Mathematical OCR (Optical Character Recognition) systems retrieve character sequences and the structure of mathematical formulae from raster images scanned from mathematical documents. In this paper a method for detecting hyperlinks, e.g. formula links, from mathematical OCR output is described. We also experimentally demonstrated the effectiveness of the method. By using the method we implemented a prototype system of a mathematical knowledge browser that helps people read mathematical articles.

1 Introduction

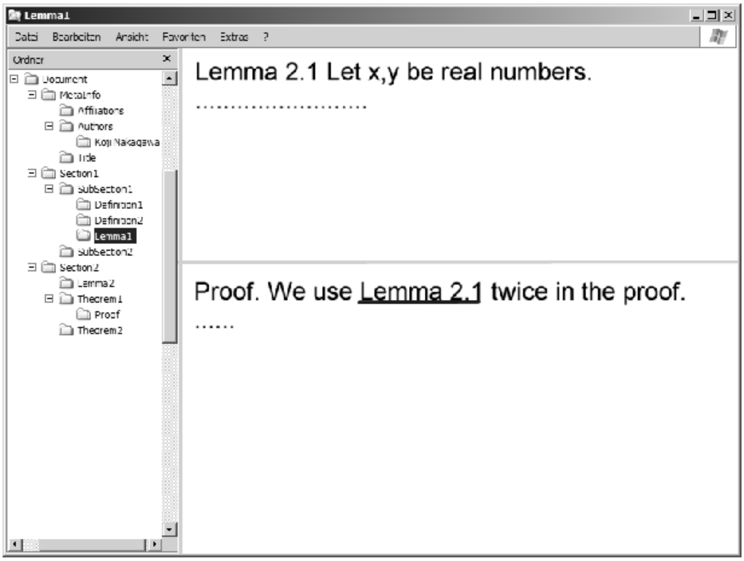
An important activity in mathematics is the reading of articles or books. Recently mathematical knowledge has started to be stored and browsed in computers, but most mathematical knowledge is still stored and browsed in printed media. Computer assistance can help the reading activity by effective functionalities, e.g. navigation with hyperlinks, which are not possible in printed media. In [5] we presented the idea of a 'mathematical knowledge browser' that helps people read mathematical articles.

For the implementation of a mathematical knowledge browser we need a method of extracting the logical structure from mathematical articles and a method of detecting hyperlinks. The method of extracting logical structure was shown to be achieved in [5], and in this paper we propose a method to automatically detect hyperlinks, which then enable effective browsing of mathematical articles. Detection of hyperlinks from OCR output for general documents was achieved in [4]. However an attempt to achieve this in mathematical documents has not been realized. Mathematical documents have more intricate structures and more types of hyperlinks, e.g. formula links, than other documents.

In Section 2 we describe the functionalities of the mathematical knowledge browser and discuss hyperlink types. By using the hyperlink detection and the automatic logical structure extraction methods we implement a prototype mathematical knowledge browser. The implemented prototype is explained in

* This work is (partially) supported by Kyushu University 21st Century COE Program, Development of Dynamic Mathematics with High Functionality, of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

M. Kohlhase (Ed.): MKM 2005, LNAI 3863, pp. 190–202, 2006.



Lemma 2.1 Let x,y be real numbers.

....

Proof. We use Lemma 2.1 twice in the proof.

......

Fig. 1. Mathematical Knowledge Browser (Sketch)

Section 3. In Section 4 the hyperlink detection method is presented. An experimental result of the proposed method is described in Section 5. Finally, we conclude in Section 6.

2 Mathematical Knowledge Browser

The mathematical knowledge browser helps people read mathematical articles. One of the inputs for this mathematical knowledge browser is the printed mathematical document. Initially, the printed mathematical document can be scanned and processed by OCR. Then the logical structure and some hyper links are automatically extracted and shown to users.

2.1 User Interface

The mathematical knowledge browser consists of three panes: structure, reference and browser panes (Fig. 1). In the structure pane located on the left side, structural information is shown as a tree that shows the logical structure and links to mathematical components such as theorems or propositions. The browser pane at the right bottom and the reference pane at the right top show the same mathematical text, but can show different positions of the text.

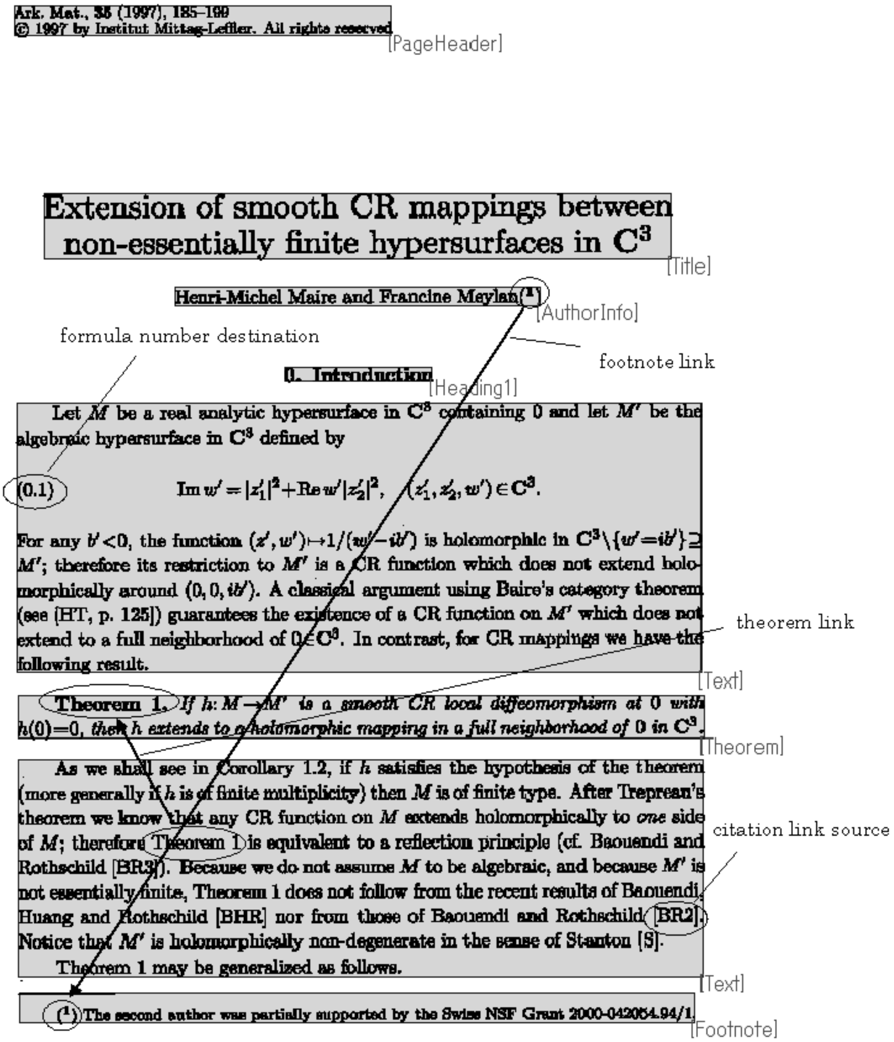
While reading an article one is often tempted to view different parts of the article at the same time, e.g. by looking back at definitions, propositions, or formulae. By clicking on a source of a hyperlink in the browser pane, the text pointed to by the hyperlink will be shown in the reference pane. By browsing this way while reading one does not lose ones attention and so can better focus on the content. For example in Fig. 1, by clicking the hyperlink ‘Lemma 2.1’ of the browser pane the content of ‘Lemma 2.1’ appears in the reference pane.

2.2 Hyperlink Types

Hyperlinks facilitate browsing activities and enhance the readability of a document by effective navigation. There are two types of hyperlinks, internal and external. In an article an internal link points to a position within the article, while an external link points to a position in another information source. Fig. 2 shows some examples of hyperlinks. Here are possible internal and external hyperlinks.

Internal Hyperlinks

- **formulae number**
  In mathematical papers formulae are often numbered for reference purposes. A formula number is located at the left or right of a formula. For example in Fig. 2, ‘(0.1)’ is an example of a formula number. Hyperlinks to the formula should be made in places where the string sequence ‘(0.1)’ appears.
- **citation**
  An article cites other documents usually by bracketed strings, e.g. ‘[12]’ or ‘[BR2]’. Detailed information of the cited documents is shown in the reference list at the end of the article. A hyperlink can be made from the place where the bracketed string is to the corresponding entry in the reference list.
- **mathematical components**
  One of distinct characteristics of mathematical articles is that there are mathematical components (e.g. Definition, Lemma and Theorem). Also in an article these mathematical components are often mainly referred to in proofs. For example, in text “By Lemma 2.4 it suffices to prove ...” the string ‘Lemma 2.4’ should link to the place where the description of ‘Lemma 2.4’ is.
- **headings (e.g. chapter, section, subsection)**
  In text, chapters or sections are sometimes referred to. For example, the sentence “This concept will be described in Section 2.” can appear in the text. Then the string ‘Section 2’ should have a link to the description of ‘Section 2’.
- **technical terms**
  If some new notions are introduced, they are named by special keywords. It is also convenient to have a hyperlink from the place where such a keyword appears to the place where the corresponding notion is introduced. However, it is difficult to recognize automatically those items. It should be solved in a different way.
- **footnote**
  A footnote identifier is usually written in the upper-script of a word at the end of a line of text. A link can be made from the footnote identifier to the footnote within the same page. See the example in Fig 2.



Ark. Mat., 35 (1997), 185–199
© 1997 by Institut Mittag-Leffler. All rights reserved

Extension of smooth CR mappings between non-essentially finite hypersurfaces in $\mathbf{C}^3$

Henri-Michel Maire and Francine Meylan(1)

0. Introduction

Let M be a real analytic hypersurface in $\mathbf{C}^3$ containing 0 and let M' be the algebraic hypersurface in $\mathbf{C}^3$ defined by

$$(0.1) \qquad \operatorname{Im} w' = |z_1'|^2 + \operatorname{Re} w' |z_2'|^2, \quad (z_1', z_2', w') \in \mathbf{C}^3.$$

For any $b'<0$, the function $(z', w') \mapsto 1/(w'-ib')$ is holomorphic in $\mathbf{C}^3 \setminus \{w'=ib'\} \supseteq M'$; therefore its restriction to M' is a CR function which does not extend holomorphically around $(0,0,ib')$. A classical argument using Baire's category theorem (see [HT, p. 125]) guarantees the existence of a CR function on M' which does not extend to a full neighborhood of $0 \in \mathbf{C}^3$. In contrast, for CR mappings we have the following result.

Theorem 1. *If $h: M \to M'$ is a smooth CR local diffeomorphism at 0 with $h(0)=0$, then h extends to a holomorphic mapping in a full neighborhood of 0 in $\mathbf{C}^3$.*

As we shall see in Corollary 1.2, if h satisfies the hypothesis of the theorem (more generally if h is of finite multiplicity) then M is of finite type. After Trepreau's theorem we know that any CR function on M extends holomorphically to *one* side of M; therefore Theorem 1 is equivalent to a reflection principle (cf. Baouendi and Rothschild [BR3]). Because we do not assume M to be algebraic, and because M' is not essentially finite, Theorem 1 does not follow from the recent results of Baouendi, Huang and Rothschild [BHR] nor from those of Baouendi and Rothschild [BR2]. Notice that M' is holomorphically non-degenerate in the sense of Stanton [S].

Theorem 1 may be generalized as follows.

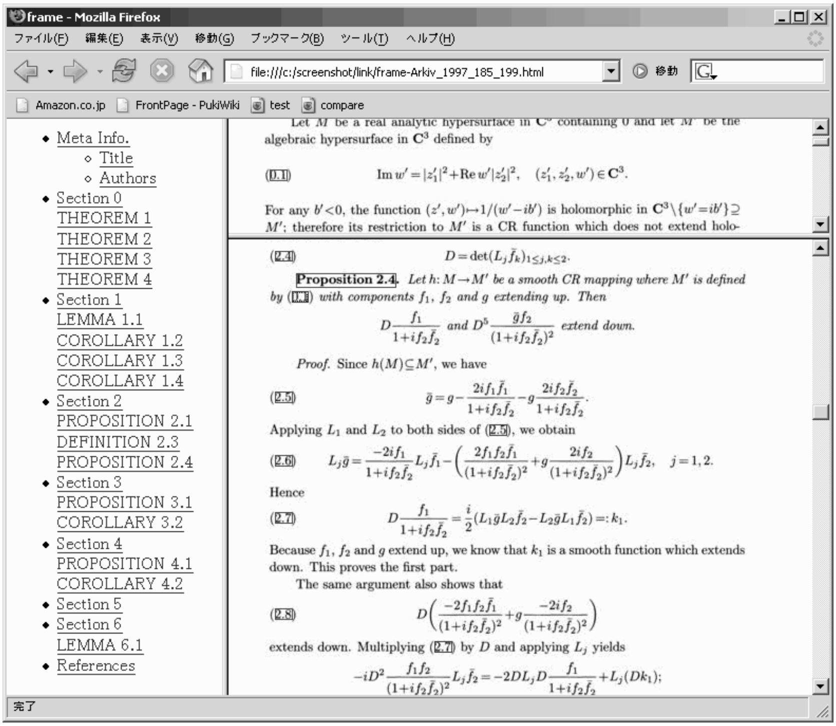
(1) The second author was partially supported by the Swiss NSF Grant 2000-042054.94/1.

Fig. 2. Examples of Internal Links

- **figure, table**
  A figure or a table can be identified by numbers separated by dots. Then a link should be made from a keyword such as ‘Figure 2.3’ or ‘Table 1.2’ to the place where the figure or the table is.

External Hyperlinks

- **common mathematical technical terms**
  There are common mathematical technical terms such as ’real number’, ’group’ or ’ring’. It would be useful to have links from these terms to online



Let M be a real analytic hypersurface in $\mathbf{C}^3$ containing 0 and let M' be the algebraic hypersurface in $\mathbf{C}^3$ defined by

$$\text{(0.1)} \qquad \operatorname{Im} w' = |z_1'|^2 + \operatorname{Re} w' |z_2'|^2, \quad (z_1', z_2', w') \in \mathbf{C}^3.$$

For any $b' < 0$, the function $(z', w') \mapsto 1/(w' - ib')$ is holomorphic in $\mathbf{C}^3 \setminus \{w' = ib'\} \supseteq M'$; therefore its restriction to M' is a CR function which does not extend holo-

$$\text{(2.4)} \qquad D = \det(L_j \bar{f}_k)_{1 \leq j,k \leq 2}.$$

Proposition 2.4. *Let $h\colon M \to M'$ be a smooth CR mapping where M' is defined by (0.1) with components f_1, f_2 and g extending up. Then*

$$D \frac{f_1}{1 + i f_2 \bar{f}_2} \text{ and } D^5 \frac{\bar{g} f_2}{(1 + i f_2 \bar{f}_2)^2} \text{ extend down.}$$

Proof. Since $h(M) \subseteq M'$, we have

$$\text{(2.5)} \qquad \bar{g} = g - \frac{2i f_1 \bar{f}_1}{1 + i f_2 \bar{f}_2} - g \frac{2i f_2 \bar{f}_2}{1 + i f_2 \bar{f}_2}.$$

Applying L_1 and L_2 to both sides of (2.5), we obtain

$$\text{(2.6)} \qquad L_j \bar{g} = \frac{-2i f_1}{1 + i f_2 \bar{f}_2} L_j \bar{f}_1 - \left(\frac{2 f_1 f_2 \bar{f}_1}{(1 + i f_2 \bar{f}_2)^2} + g \frac{2i f_2}{(1 + i f_2 \bar{f}_2)^2} \right) L_j \bar{f}_2, \quad j = 1, 2.$$

Hence

$$\text{(2.7)} \qquad D \frac{f_1}{1 + i f_2 \bar{f}_2} = \frac{i}{2} (L_1 \bar{g} L_2 \bar{f}_2 - L_2 \bar{g} L_1 \bar{f}_2) =: k_1.$$

Because f_1, f_2 and g extend up, we know that k_1 is a smooth function which extends down. This proves the first part.

The same argument also shows that

$$\text{(2.8)} \qquad D \left(\frac{-2 f_1 f_2 \bar{f}_1}{(1 + i f_2 \bar{f}_2)^2} + g \frac{-2i f_2}{(1 + i f_2 \bar{f}_2)^2} \right)$$

extends down. Multiplying (2.7) by D and applying L_j yields

$$-i D^2 \frac{f_1 f_2}{(1 + i f_2 \bar{f}_2)^2} L_j \bar{f}_2 = -2 D L_j D \frac{f_1}{1 + i f_2 \bar{f}_2} + L_j (D k_1);$$

Fig. 3. Screenshot of Prototype Implementation

mathematical dictionaries, e.g. MathWorld[1], so that one can easily understand or recall the notions without having to physically look up books. Since it may happen that a common term can mean different concepts in different areas of mathematics, the link destination should be search pages of mathematical dictionaries.

- **reference linking**
  Cited articles are listed in a reference list. It is possible to create hyperlinks from articles of the reference list to the information of the cited articles. The technology to identify articles is called 'reference linking'[1, 3]. The destination of such a hyperlink can be an entry of a mathematical review site[2] or the place where the article is.

3 Prototype Implementation

We implemented a prototype of our mathematical knowledge browser using ordinary functionalities of standard web browsers (Fig. 3). Here the article is shown as a sequence of bitmap images. Sources and destinations of hyperlinks are shown

[1] http://www.mathworld.com

[2] http://www.ams.org/mathscinet

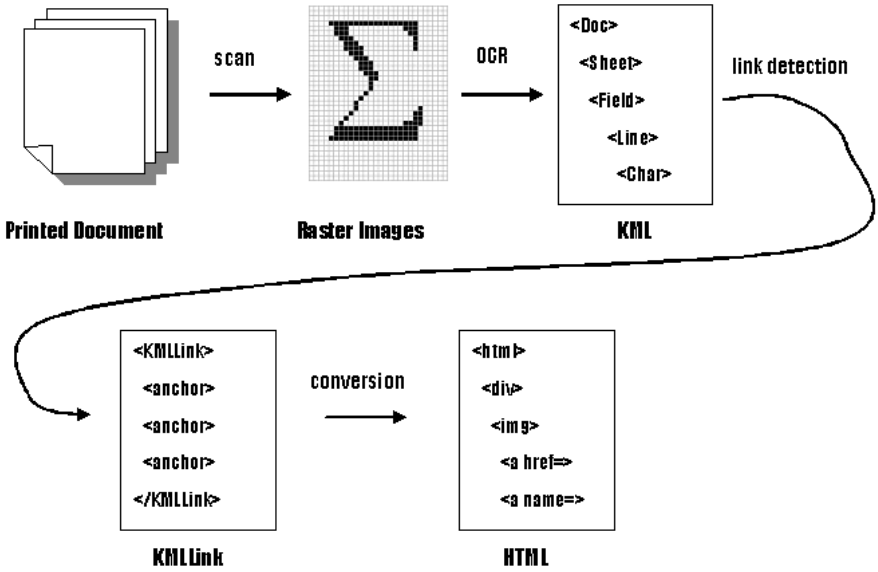


Fig. 4. Process Flow of Prototype Implementation

as surrounding boxes overlapping the bitmap images. The source of a hyperlink is colored red, and the destination green.

The process flow of the prototype implementation is shown in Fig. 4. At first, printed materials are scanned and then converted into raster image files. Then these images are processed by an OCR engine. We use an integrated OCR system for mathematical documents called INFTY[3][6]. INFTY reads the scanned page images of a mathematical document and provides character recognition results. One of the important characteristics of INFTY is that it can recognize two-dimensional mathematical expressions. The recognition result can be saved in a XML format called KML, which includes the results of logical structure analysis. A KML file is analyzed by a link detection program that produces the result in KMLLink format. From KMLLink and KML files some HTML files can be produced by a conversion program and browsed by ordinary web browsers.

The contributions this paper makes are the link detection program and the program for conversion into HTML. These programs are written in Python[8], which is a script language that conveniently handles XML.

3.1 KML: An OCR Result Format with Meta-information and Logical Structure

INFTY produces output in a XML format called KML. For example, Fig. 5 shows the output results in KML for the scanned image shown in Fig. 2. The top element is ‘Doc’ which contains some ‘Sheet’ elements representing pages. A ‘Sheet’

[3] INFTY is freely available from `http://www.inftyproject.org/en/`

```
<Doc version="1.1" language="English" ...>
 <Sheet id="1" doc_file_name="Arkiv_1997.kml"
   image_file_name="Arkiv_1997_185.tif" height="4438" width="3015" ...>
  <Area rect="148,129,1801,266" id="1" ...>
   <Text rect="148,129,1801,266" tag="PageHeader" ...>
    <Field base_char_size="16,30,13,41" sub_char_size="11,20,9,28">
     <Line id="1" rect="148,129,1086,195">
      <Char code="0141" rect="148,133,195,180" ...>A</Char>
      <Char code="0172" rect="200,151,223,180" ...>r</Char>
      ...
     </Line>
     <Line id="2" rect="149,200,1801,266">
      ...
  </Area>
  <Area rect="279,948,3022,1239" id="2" ...>
   <Text rect="279,948,3022,1239" tag="Title" ...>
   ...
  </Area>
   ...
 </Sheet>
 <Sheet id="2" doc_file_name="Arkiv_1997.kml"
   image_file_name="Arkiv_1997_186.tif" height="4432" width="3002" ...>
  <Area rect="229,169,326,215" id="1">
   <Text rect="229,169,326,215" tag="PageNumber" ...>
    <Field base_char_size="16,28,12,39" sub_char_size="11,19,8,26">
     <Line id="1" rect="229,169,326,215">
   ...
   </Text>
  </Area>
  <Area rect="1088,168,2358,228" id="2">
   <Text rect="1088,168,2358,228" tag="PageHeader" ...>
    <Field base_char_size="16,29,12,40" sub_char_size="11,20,8,27">
     <Line id="1" rect="1088,168,2358,228">
  ...
  </Area>
  <Area rect="231,405,3224,701">
   <Text tag="Theorem">
    <Field>
     <Line id="1" rect="392,405,3224,486">
      <Char code="2154" rect="392,410,452,467" bold="1"...>T</Char>
      <Char code="2168" rect="458,409,506,467" bold="1"...>h</Char>
      ...
  </Area>
 <CharInfo>... </CharInfo>
</Doc>
```

Fig. 5. Example of KML Output from an INFTY OCR Engine

```
<KMLLink verion="1.0" page="23">
<anchor type="destination" kind="mathcomp" label="THEOREM 2.1"
          page="3" rect="106,3290,137,3340" />
...
<anchor type="source" kind="mathcomp" label="THEOREM 2.1"
          page="10" rect="2462,1734,2495,1789" />
...
</KMLLink>
```

Fig. 6. Example of KMLLink Format

element contains some ‘Area’ elements whose positions and sizes are indicated by ‘rect’ attributes. The value of the ‘rect’ attribute "$left$,up,$right$,$down$" indicates the positions of left, up, right, down borders, respectively, of the rectangle. An ‘Area’ element contains a ‘Text’ element having a ‘Field’ element. A ‘Field’ element has several ‘Line’ elements that again have several ‘Char’ elements.

To satisfy the need to put additional information for meta-information and logical structure, the ‘tag’ attribute for the ‘Text’ element exists to represent the type of the text field. The values of the ‘tag’ attribute are ‘PageHeader’, ‘PageNumber’, ‘Caption’, ‘Title’, ‘AuthorInfo’, ‘AbstractHeader’, ‘Abstract’, ‘Keywords’, ‘Heading1’, ‘Heading2’, ‘Heading3’, ‘Heading4’, ‘Heading5’, ‘Text’, ‘Bibitem’, ‘Definition’, ‘Axiom’, ‘Theorem’, ‘MainTheorem’, ‘Proposition’, ‘Corollary’, ‘Lemma’, and ‘Footnote’.

3.2 KMLLink: Link Description Language for KML

The link detection program takes a KML file as input and produces results in the KMLLink format. Fig. 6 shows an example in KMLLink. The top element is ‘`KMLLink`’ that contains only ‘`anchor`’ elements. There are two types of ‘`anchor`‘: ‘`source`’ and ‘`destination`’ specified by the ‘`type`’ attribute. The ‘`kind`’ attribute takes one of three values: ‘`citation`’, ‘`formula`’, or ‘`mathcomp`’ (mathematical component). The ‘`page`’ and ‘`rect`’ attributes specify a rectangle in a page. The ‘`label`’ attribute specifies the identifier of a link. A pair of ‘`source`’ and ‘`destination`’ anchors that have the same label indicate a link. For example, in Fig. 6 a pair of two anchors indicate a ‘`Theorem 2.1`’ link from a place in page ‘`10`’ to a place in page ‘`3`’.

3.3 Conversion from KMLLink to HTML

The conversion program takes a KMLLink file and a KML file as input, and produces three HTML files:

- frame file,
- navigation file,
- content file.

The frame file forms the outline that contains three panes by using the ‘FRAME’ element of HTML. The content of the navigation pane is described

```
<div style="position:relative;top: 0px; left:0px;...">
 <img src="/images/InvM_1970_121_134-0.jpg">
 <a href="#Theorem 2.4" style="position:absolute;
                      left:397px; top:262px; width:5px; height:9px;
                      border: 1px solid red;" target="reference"></a>
 ...
 <a name="Theorem 2.4" style="position:absolute;
                       left:27px; top:528px; width:386px; height:80px;
                       border: 1px solid green;"> ...</a>
</div>
```

Fig. 7. HTML Realization of Hyperlinks over a Raster Image for a Page

in the navigation file. Production of the navigation file needs the result of the logical structure analysis that is stored in the KML file. Both the reference and the browse panes show the same content file in which scanned images are vertically allocated and browsed by scroll bars. (As images used for OCR are large for browsing, the size of scanned images is decreased by 15%.)

It is also possible to show the content in MathML(+HTML), since the INFTY OCR engine can recognize mathematical formulae. However, we chose raster images for showing pages because there are some miss-recognitions by OCR and it is not always the case that web browsers can display MathML properly.

In a content file, destinations and sources of hyperlinks are indicated by surrounding boxes that are realized by specifying the '`style`' attribute of HTML. For example, a page is represented in Fig. 7. An image is shown by the '`img`' element. The source of a hyperlink is realized by the '`a`' element with the '`href`' attribute. The '`target`' attribute specifies the target window `"reference"`, which represents the reference pane. The destination of a hyper link is realized by the '`a`' element with the '`name`' attribute. In the '`style`' attribute, the positions and sizes of surrounding boxes are specified by '`left`','`top`', '`width`', and '`height`' with '`border`'.

4 Automatic Link Detection Method

In this paper we focus on the detection of three internal link types: formula, citation, and mathematical components; because these are especially useful in browsing. Other links will be the subject of future work. Automatic link detection can be achieved by looking for specific string patterns. A link is specified by its source and its destination. In most cases, the string pattern of the source and the destination of a link are the same.

Although we can not expect string patterns that will work for all articles, in this paper we use fixed patterns that should work in most cases. Fig. 8 shows the fixed patterns (regular expressions) used for detecting destinations and sources of links. Basically the algorithm looks for these fixed patterns line by line, and decides whether what it finds is a destination or a source.

For there to be more accurate recognition of links, there needs to be some mechanisms by which one can specify the string patterns of links. For example, in

kind	regular expression	example
formula	\([0-9]+(\. ?[0-9]+)*\'?\)	(2) (1.2) (3')
citation	\[([^\[^\]]*)\]	[2] [Mar80,Buc99]

mathcomp	Theorem([0-9]+(\. ?[0-9]+)*'?| [a-zA-Z]+|)
	Lemma([0-9]+(\. ?[0-9]+)*'?| [a-zA-Z]+|)
	...
		Theorem 3.2
		Lemma II

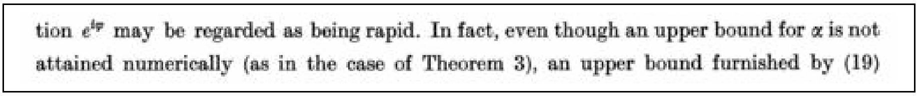
Fig. 8. Used Regular Expressions

[4] they define a link specification language called LITHP (Link Type description language for HyperText Processing) by which one can define link patterns.

Here the detailed algorithm is explained for each link.

4.1 Formula Link

Link Destination Detection. Sometimes a formula has a label written in a parenthesized number, or numbers separated by dots, e.g. '(2)' or '(2.1)' or '(2.2.3)', at the left or right of the formula. However all such labels do not necessarily become formula link destinations. For example:



tion $e^{i\varphi}$ may be regarded as being rapid. In fact, even though an upper bound for α is not attained numerically (as in the case of Theorem 3), an upper bound furnished by (19)

Here '(19)' must not be recognized as a formula link destination. To avoid this problem, only the first occurrences of such labels are considered to be destinations of formula links, because in most cases these labels that are not destinations come after the formula is labeled.

Link Source Detection. All strings that match the regular expression for 'formula' in Fig. 8, and are the same as link destinations' labels are link sources.

4.2 Citation Link

Link Destination Detection. Destinations of citation links can be detected from the reference section. Usually an reference entry starts with either a bracket string (e.g. '[Buc2004]') or numbers with a dot (e.g. '12.').

Link Source Detection. A citation link source is usually written in the form of '[$str_1, \cdots, str_n$]'. However all $str_1, \cdots, str_n$ do not always indicate the source of citation links. For example, '[8, Theorem 3]' indicates 'Theorem 3' of the paper that is indicated by the citation number '8'. The label '[7, pp. 38]' indicates that it refers to the 38th page of the article cited by the number '7'. Another example is an interval notation '[a,b]'.

Here, the way to distinguish is that strings occurring in the reference list (link destinations) are considered to be citation sources. Then the examples 'Theorem 3' and 'pp. 38' are not strings of a citation source. Additionally after the first occurrence of a non-citation string, all such strings are considered to be non-citation strings. Namely suppose we have '$[str_1, \cdots, str_{i-1}, str_i, \cdots, str_n]$' and the strings from str_1 to str_{i-1} appear in the reference list, but str_i does not appear in the reference list, the strings from str_i to str_n are not considered to be citation sources. For example, let us consider the case '[8, Theorems 3, 4]'. Suppose '4' and '8' appear in the reference list. In this case '8' is considered to be a citation label, but '4' is not because a non-citation string 'Theorems 3' appears before '4'.

4.3 Mathematical Component Link

Link Destination Detection. Destinations can easily be detected after logical structure extraction, because in KML the 'Text' elements are tagged by keywords, e.g. 'Theorem'. The beginnings of such 'Text' elements are mathematical component link destinations. For example, in Fig. 2 'Theorem 1' is a mathematical component link destination.

Table 1. Experimental Result of Detecting Hyperlinks

paper ID	formula		citation		math. comp.	
	source	dest.	source	dest.	source	dest.
ActaM_1970_37_63	92/92[2]	54/55[1]	18/18[1]	7/7[0]	54/54[2]	16/16[0]
ActaM_1998_283_305	0/0[23]	0/0[5]	33/33[0]	12/12[0]	38/38[3]	30/34[4]
AIF_1970_493_498	0/0[0]	0/0[1]	6/6[0]	2/2[0]	4/7[0]	1/1[0]
AIF_1999_375_404	4/4[18]	1/4[3]	18/18[0]	12/12[0]	44/46[0]	34/34[0]
AnnMS_1971_157_173	2/2[3]	0/2[3]	17/18[0]	6/6[0]	6/12[3]	11/11[0]
AnnM_1970_550_569	55/55[0]	29/29[0]	24/24[0]	20/20[0]	40/46[0]	6/6[0]
Arkiv_1971_141_163	0/0[0]	3/3[0]	24/24[0]	7/7[0]	41/42[0]	24/24[0]
Arkiv_1997_185_199	53/53[4]	42/42[2]	24/24[0]	12/12[0]	30/32[2]	16/16[0]
ASENS_1970_273_284	0/0[0]	0/0[0]	32/32[0]	14/14[0]	9/9[2]	7/7[0]
ASENS_1997_367_384	0/0[13]	1/1[2]	33/33[0]	15/15[0]	34/41[3]	18/18[0]
BAMS_1971_157_159	7/7[0]	9/9[0]	7/7[0]	6/6[0]	6/6[0]	3/3[0]
BAMS_1971_160_163	0/5[0]	0/3[0]	6/6[0]	6/6[0]	1/1[0]	6/6[0]
BAMS_1974_1219_1222	0/0[0]	0/0[0]	6/6[0]	2/2[0]	0/4[0]	9/9[0]
BAMS_1998_123_143	0/0[0]	0/0[0]	113/113[0]	48/48[0]	33/35[0]	34/34[0]
BSMF_1970_165_192	18/18[0]	15/15[0]	71/71[0]	8/8[0]	41/50[0]	16/16[0]
BSMF_1998_245_271	50/50[0]	34/34[0]	41/41[0]	21/21[0]	37/48[0]	19/20[0]
InvM_1970_121_134	46/46[1]	19/19[1]	30/30[0]	7/7[0]	0/0[2]	2/2[0]
InvM_1999_163_181	31/31[26]	18/18[5]	0/18[0]	0/6[0]	0/3[21]	17/19[0]
JMKU_1971_181_194	16/16[8]	14/14[0]	10/11[0]	9/9[0]	16/20[10]	8/12[3]
JMKU_1971_373_375	0/0[4]	0/0[6]	6/6[0]	4/4[0]	1/1[1]	3/3[0]
JMS_1975_281_288	10/10[0]	6/6[0]	18/18[0]	11/11[0]	19/19[0]	16/16[0]
JMS_1975_289_293	3/3[0]	6/6[0]	6/6[0]	3/3[0]	1/2[0]	3/3[0]
JMS_1975_497_506	43/43[0]	43/43[0]	19/19[0]	10/10[0]	4/5[1]	7/7[0]
KJM_1999_17_36	42/42[2]	27/27[5]	22/22[0]	7/7[0]	40/40[5]	21/24[1]
MA_1977_275_292	38/39[0]	29/30[0]	24/24[0]	14/14[0]	11/13[2]	12/12[0]
MA_1999_175_196	39/39[1]	37/37[2]	36/36[0]	27/27[0]	13/15[1]	6/6[0]
TMJ_1973_317_331	0/0[0]	0/0[0]	19/22[0]	12/12[0]	17/17[0]	11/11[0]
TMJ_1973_333_338	0/0[0]	0/0[0]	6/9[0]	6/6[0]	6/6[0]	5/5[0]
TMJ_1990_163_193	109/116[53]	41/44[6]	57/65[0]	31/31[0]	51/64[0]	25/26[0]
Sum	658/671[158]	428/441[42]	726/760[1]	339/345[0]	597/676[58]	386/401[8]
	98.1%	97.1%	95.5%	98.3%	88.3%	96.3%

Link Source Detection. Strings that are the same as the strings of mathematical component link destinations are mathematical component link sources.

5 Experiment

To show the effectiveness of the link detection method we set up an experiment. A large-scale database of mathematical articles [6, 7] stored in KML was utilized. We randomly chose 29 English articles on pure mathematics (issued in 1970 - 1999) from different journals. Basically, an old and a new paper are chosen for each journal.

From the database in KML we made a correct KMLLink database and initiated the experiment (Table 1). A table entry is in the form '$success/all[excess]$'. 'all' is the number of all correct elements. '$success$' is the number of elements successfully detected by the method presented in this paper. '$excess$' is the number of elements excessively detected by the method. Note that the decrease of the number of '$excess$' means better result. In total, the result achieved a 95.1% success rate with 267 excessively detected elements, which was 8.1% of all correct elements.

6 Conclusion

A method to detect several types of hyperlinks from printed mathematical documents was proposed. Using the method, we implemented a prototype mathematical knowledge browser. The authors believe that automatically detected hyperlinks make browsing of mathematical articles more effective. We intend to improve our hyperlink detection method and apply the improved version to larger scale databases.

In general, the style assumptions described in this paper do not work for mathematical documents whose styles are completely different. For example for citations some articles use a parenthesized form, e.g. '(Buc 2000)'. To adapt the system to such cases, a mechanism by which one can specify string patterns by regular expressions is needed. With such a mechanism, the system will work for exceptional cases.

For the prototype implementation of the mathematical knowledge browser we used standard web browsers, but for greater functionality we will need to implement standalone software. The following improvements can be considered for our mathematical knowledge browser:

- Elaborate Search
  Automatic detection of internal hyperlinks of technical keywords is a difficult task. A practical solution would be to provide an elaborate search functionality in the browser. By selecting a keyword in the browser and pressing a button, all words that are the same as the keyword in the paper will be marked and they can then be browsed sequentially.

– Showing Overview
  An article can be shown in an overview mode. For example, it is possible to show only the numbered mathematical formulae that appear in an article. In this way, one may be able to get the general idea of the paper.

Mathematical knowledge needs to be stored in a content-base format rather than a presentation-base format so that it can be used for various purposes. Mathematical knowledge should be store in a higher level format, because at this higher level practical usage is enhanced. However, currently most mathematical knowledge is stored in printed media and the situation will not change much without some action been undertaken. The technologies presented here support converting lower level formatted knowledge into higher level formatted knowledge. We hope that in the future people will store mathematical knowledge in a content-base format such as OMDoc[2].

References

1. Donna Bergmark. Automatic extraction of reference linking information from online documents. Technical report, 2000. CSTR 2000-1821.
2. M. Kohlhase. OMDoc: An Infrastructure for OpenMath Content Dictionary Information. *SIGSAM Bulletin (ACM Special Interest Group on Symbolic and Algebraic Manipulation)*, 34(2):43–48, 2000.
3. Steve Lawrence, C. Lee Giles, and Kurt Bollacker. Digital libraries and Autonomous Citation Indexing. *IEEE Computer*, 32(6):67–71, 1999.
4. A. Myka and U. Güntzer. Automatic Hypertext Conversion of Paper Document Collections. In *Digital Libraries: Current Issues, Digital Libraries Workshop, Newark, NJ, USA, May 19-20, 1994, Selected Papers*, volume 916 of *Lecture Notes in Computer Science*, pages 65–90. Springer, 1995.
5. K. Nakagawa, A. Nomura, and M. Suzuki. Extraction of Logical Structure from Articles in Mathematics. In A. Trybulec A. Asperti, G. Bancerek, editor, *Mathematical Knowledge Management, Third International Conference, MKM 2004, Bialowieza, Poland, September 19-21*, volume 3119 of *Lecture Notes in Computer Science*, pages 276–289. Springer, 2004.
6. M. Suzuki, F. Tamari, R. Fukuda, S. Uchida, and T. Kanahori. INFTY — An Integrated OCR System for Mathematical Documents. In *ACM Symposium on Document Engineering (DocEng '03), Grenoble, France, Nov. 20-22*, 2003.
7. S. Uchida, A. Nomura, and M. Suzuki. Quantitative analysis of mathematical documents. *International Journal on Document Analysis and Recognition*, 2005. ISSN: 1433-2833 (Paper) 1433-2825 (Online).
8. Guido van Rossum. *Python Reference Manual.* Python Software Foundation, release 2.4.1 edition, March 2005.