Chapter 8

Document Recognition for a Digital Library

Sargur N. Srihari*, Stephen W. Lam† and Jonathan J. Hull‡,

8.1 Introduction

An important part of the development of a digital library [DL] is the transformation of digital images of documents into an ASCII representation. Although newer entries in a conventional library may be in electronic form already, the majority of library archives is still in printed form. The latter needs to be automatically recognized, represented in a form suitable for information retrieval [IR], and integrated.

The transformation of digital images of print documents into text-searchable form is popularly known as document image understanding [DIU], and includes the processes of *document layout understanding*, *text recognition* and *logical linking*. All of these steps are crucial for IR.

Document layout understanding [DLU] includes the processes of *segmenting* multiple page documents into text, graphics, and halftone images, *labeling* these elements into meaningful entities (such as title, author, section header, etc.), and *grouping* these entities on the same page or from different pages into logical units. The text regions are processed by an optical character reader [OCR] which produces ASCII. A DLU system which uses an adaptive approach can be applied to a variety of documents.

An important part of document recognition is the recognition of the images of text. Current OCRs operate mainly on a character-by-character basis and at times a lexicon is used to postprocess the results. This provides reasonable

method for interpreting the results of a text recognition algorithm based on concepts from IR.

The logical linking of document elements is important for IR. Figures, tables, equations and bibliographies can be automatically retrieved when the text that matches a user query has explicit or implicit references to those elements. This is difficult when the images are degraded, since the figure and table numbers may consist of only a single digit, probably of poor quality. Our solution to the linkage problem detects the reference links in the text and locates the referenced elements. This will allow the user to retrieve specific areas of the document which are related to the query rather than retrieving the entire document.

The following describes the three areas mentioned above in more detail.

8.2 Adaptive Document Layout Understanding

Recognition problems occur when there are a variety of page layout, print quality and contextual differences. This research focuses on improving the methods used to process large volumes of documents which are found in a typical library. The DLU process consists of four stages: (i) skew correction, (ii) block segmentation, (iii) block classification, and (iv) layout understanding. Block segmentation and classification can be broadly grouped into a single stage called zoning. Since most of the techniques developed for zoning require content of a document with proper alignment, a skewed document image has to be corrected prior to the zoning stage.

8.2.1 Skew Correction

A fast skew correction algorithm based on the estimation of the projection profile complexity [130] has been developed. Projection profiles on the foreground pixels are generated at several orientations ($\pm 10°$ are used). It detects the orientation of the white gaps between text lines (see Figure 8.2.1). The orientation of the profile with the largest variance will be selected as the skew angle of the document. A document image is first partitioned into several small regions. Only those text regions will be used to determine the document orientation. It can also process documents with text printed in a vertical direction such as Chinese and Japanese documents (in this case, $80° - 100°$ are used). It can accurately estimate the skew within 0.5 degree and takes 3 CPU seconds on a SUN SPARC2 for a letter-sized document.

8.2.2 Block Segmentation and Classification

Layout analysis starts with block segmentation which decomposes the digital image of a document page into regions. The process locates large st[illegible]

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

a tilted axis. The recognition result is shown in Fig. 21(b). (We overlayed a grid of the range image of the model, which was transformed by the resulting transformation on top of Fig. 21(a).) In Fig. 21(c) and (d), we show the final hypotheses, which lead to the result in Fig. 21(b). Hypotheses 2 and 3 and 18 and 19 are overlaid.

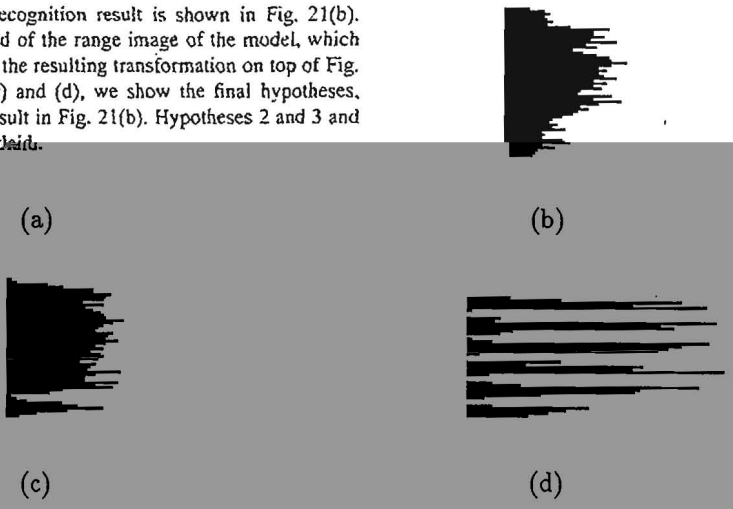


Figure 8.1: Profiles of a text block generated at different projection orientations: (a) image contains skewed text, (b) profile projected at -3°, (c) profile projected at 0°, and (d) profile projected at 3°.

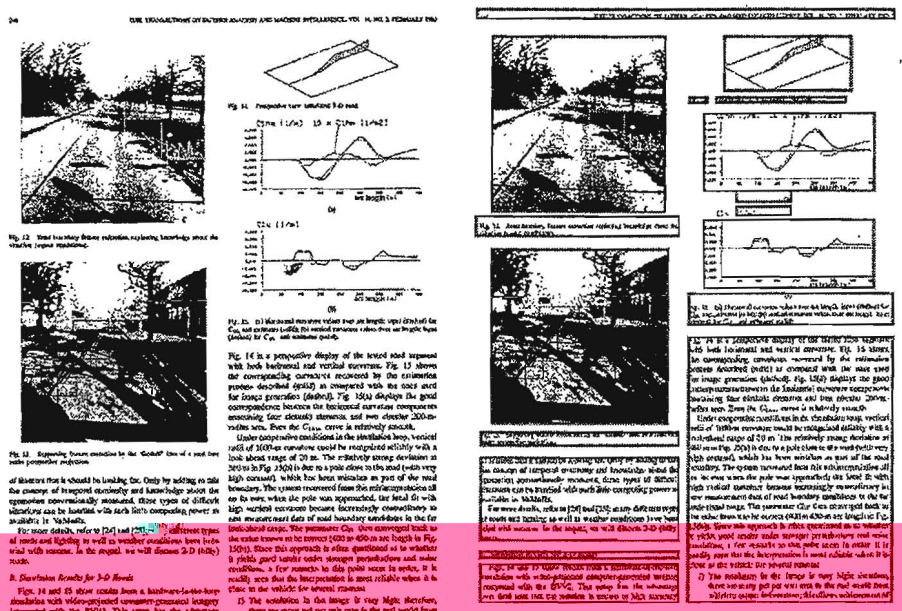
A region must be bound by two horizontal and vertical background boundaries. Criteria of considering a white stream as region boundary are derived from the analysis of all the white streams in the page image. Non-boundary white streams, which are usually smaller in size, separate elements of the same region such as those between text lines. This global statistical approach does not need to predefine the size of the white streams, which varies from page to page. This local adaptive method increases the robustness of the segmentation process. Figure 8.2.2 shows the regions of a document after block segmentation.

The next step is classifying a region into one of the structural categories such as text, line drawings, tables and photographs. The classification of a region is performed by matching a set of features extracted from the region against the predefined reference features of a category. This approach allows the addition of new categories only if they contain distinctive features. Therefore, it has great flexibility in handling various types of documents.

8.2.3 Layout Understanding

Region labeling is to assign a region a document specific semantic tag. Different documents have different sets of tags, *e.g.*, a text region on a technical journal is labeled as title while a text region on a newspaper page is labeled as headline. Therefore, this process is knowledge-driven. The knowledge contains information about all possible semantic tags on a page and their spatial relationships.

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.



(a) (b)

Figure 8.2: Result of block segmentation. (a) Original document image. (b) Regions located by the block segmenter.

Logical grouping is invoked when all the pages of the document have gone through the previous stages. The physical partition of document elements becomes transparent to the grouping process. The grouping is guided by the knowledge about the content of the whole document. For example, the knowledge for analyzing a technical publication defines all the essential components of a journal. The group process locates all these components according to given specifications. The results of the grouping process contain the logical units and their physical locations represented in SGML format.

The tasks of labeling and grouping can be combined into a single process called layout understanding. A system can be categorized as either "closed" or "open", depending on whether it is designed for a particular class of document. Most systems in use today are closed systems, *i.e.*, they are designed for some specific documents (such as forms, specific journals and business letters). They cannot easily be adapted to other types of documents.

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

An open system architecture has been developed at CEDAR. It is designed for processing multi-page documents, *i.e.*, it generates a logical interpretation of a document by combining information on different pages of the document. The architecture, (see Figure 8.2.3), consists of three components: control, knowledge base and tool box. The *Control* possesses some general purpose document analysis strategies and the *Tool Box* contains a set of generic document image processing tools that are applicable to different documents. The system has no prior knowledge about document domain. The use of strategies and tools relies solely on the knowledge of the document of interest defined in the *Knowledge Base*. Since document-specific knowledge is not part of the system, it can be viewed as input (in addition to the document images) to the system. A prototype system based on this architecture has been developed to process a variety of documents such as forms, IEEE journals and postal mailpieces [154, 153]. The research focus is on how to use knowledge to adjust the system reading strategies for handling different types of library documents.

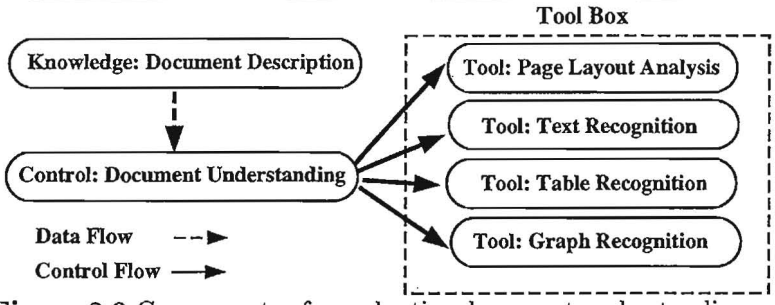


Figure 8.3 Components of an adaptive document understanding system.

8.3 Text Recognition Using Document Context

The recognition of word images is a solution to text recognition in which images of text are transformed into their ASCII equivalent. Word recognition algorithms are an alternative to traditional character recognition techniques which rely on the segmentation of a word into characters. This is sometimes followed by a postprocessing step that uses a dictionary of legal words to select the correct choices.

Errors in the output of a word recognition system can be caused by several sources. When a noisy document image is input, the top choice of a word recognition system may be correct relatively infrequently. However, the ranking of the dictionary may include the correct choice among its top N guesses (N=10, for example) in nearly 100% of the cases.

Solutions for improving the performance of a text recognition system have utilized the context of the language in which the document was written. An observation about context beyond the individual word level that is used here

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

concerns the vocabulary of a document. Even though the vocabulary over which word recognition is computed may contain 100,000 or more words, a typical document may actually use fewer than 500 different words. Thus, higher accuracy in word recognition is bound to result if the vocabulary of a document could be predicted and the decisions of a word recognition algorithm were selected from that limited set only.

This chapter discusses a methodology to predict the vocabulary of a document from its word recognition decisions. The N best recognition choices for each word are used in a probabilistic model for information retrieval to locate a set of similar document in a database. The vocabulary of those documents is then used to select the recognition decisions from the word recognition system that have a high probability of correctness. Those words could then be used as "islands" to drive other processing that would recognize the remainder of the text. A useful side effect of matching word recognition results to documents from a database is that the topic of the input document is indicated by the titles of the matching documents from the database. The algorithmic framework discussed in this chapter is presented in Figure 8.3. Word images from a document are input. Those images are passed to a word recognition algorithm that matches them to entries in a large dictionary. *Neighborhoods* or groups of words from the dictionary are computed for each input image. The neighborhoods contain words that are *visually* similar to the input word images.

A matching algorithm is then executed on the word recognition neighborhoods. A subset of the documents in a pre-classified database of ASCII text samples are located that have similar topics to the input document. The hypothesis is that those documents should also share a significant portion of their vocabulary with the input document.

Entries in the neighborhoods are selected based on their appearance in the matching documents. The output of the algorithm are words that have an improved probability of being correct based on their joint appearance in both the word recognition neighborhoods as well as the matching documents. These are words that are both visually similar to the input and are in the vocabulary of the documents with similar topics.

8.3.1 Experimental Investigation

The word decision selection algorithm discussed in this chapter was demonstrated on the Brown corpus[151]. The Brown corpus is a collection of over one million words of running text that is divided into 500 samples of approximately 2000 words each. The samples were selected from 15 subject categories or genres and the number of samples in each genre was set to be representative of the amount of text in that subject area at the time the corpus was compiled.

Testing Data

One of the samples in the Brown corpus was selected as a test document to demonstrate the algorithm presented in this chapter. This sample is denoted

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

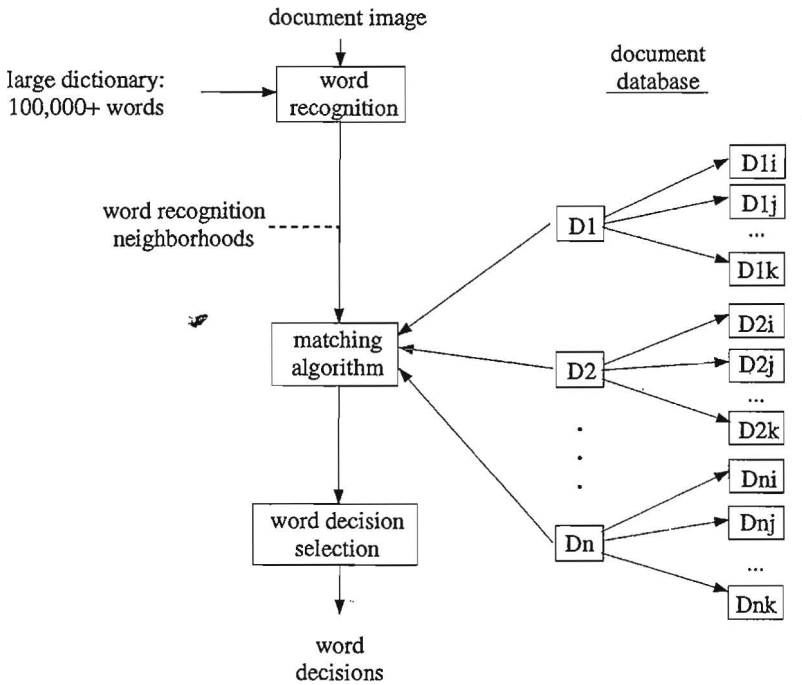


Figure 8.4 A proposed word matching algorithm.

G02 (the second sample from genre G: *Belles Lettres* and is an article entitled *Toward a Concept of National Responsibility*, by Arthur S. Miller that appeared in the December, 1961 edition of the Yale Review.

There are 2047 words in the running text of G02. After removing stop words and proper nouns, there were 885 words left. Raster images were generated for those words with a postscript-to-bitmap generation technique. This was done to provide test data for a recognition algorithm that would compute neighborhoods of visually similar words for each of the 885 input words. The stop words and proper nouns were excluded from the test data set since it was assumed that algorithms existed to find those words in a document image.

Neighborhoods were generated for each word in G02 with a word shape calculation in which a feature vector that describes the global characteristics of a word is compared to similar feature vectors for each word in a dictionary [125]. A ranking of the dictionary results in which words that are visually similar to an input image are ranked close to the top. For the experimentation discussed here, the approximately 53,000 unique words that occur in the Brown Corpus

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

were placed in the dictionary.

The ten most visually similar dictionary words were calculated for each input word. This provided 8850 neighbors overall. The word shape calculation had performance of 87% correct in the top choice and 99% correct in the top ten choices.

Training Data and Results

The training data for the matching process and the word decision selection algorithm was the other 499 samples in the Brown corpus besides G02. The document matching algorithm described earlier was used to rank the other 499 samples for their similarity to G02.

The ability of the most similar samples determined by the matching procedure to select the correct word decisions from the neighborhoods was tested under three noise conditions using three selection criteria.

Noise was introduced in the word recognition output to test the tolerance of the decision selection procedure to imperfect input. A uniform random number generator was used to select a given number of neighborhoods from among those that had the correct decision as the first choice. The second choice was substituted for the first, thus providing a neighborhood that contained a visually similar, but incorrect, word in the first position. A 24% error rate was simulated by applying the above procedure to 93 of the 769 neighborhoods that were correct (i.e., the top choice of the recognition algorithm was correct). A 30% error rate was introduced with a similar method.

The top choices of the recognition algorithm were filtered by comparing them to the most similar samples and retaining the words that occurred in those samples. The three selection criteria that were tested included *overall* performance in which all the top recognition choices in G02 that occurred anywhere in the similar samples were retained.

The *G02-nouns* condition refers to the case where only the top choices for the nouns in G02 that matched any of the nouns in the similar samples were retained. The application of this selection criteria in a working system would assume the presence of a part-of-speech (POS) tagging algorithm that would assign POS tags to word images.

In the *matching-nouns* condition, only the nouns in the similar samples were used to filter the top recognition choices. This case was explored because nouns may be considered carrying more information about the content of a text passage rather than verbs or words with other parts of speech. Thus, the co-occurrence of nouns in two documents about similar topics should be due less to chance than other word types.

The results of word decision selection when applied to the original word recognition output (with 13% error at the top choice) are summarized in Table 8.1. When all the words in the most similar sample (J42) were matched to the top recognition decisions for G02 (top left entry in Table 8.1), it was discovered that 251 of those top decisions also occurred in J42. Of those, only nine words were erroneous matches. This corresponds to an error rate of about 4%. In other

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

words, the correct rate for 28% of the input words was raised to 96% from the 87% provided by the word recognition algorithm alone.

The other results show that as more of the similar samples are used to filter the word recognition output, a progressively higher percentage of the eligible neighborhoods are included and the correct rate remains stable. For example, in the *overall* condition using the four most similar samples, 441 of the 885 (50% input words were effectively recognized with a correct rate of 97%. The results for the *G02-nouns* matching condition show that up to 26% of the input can be recognized with a 99% correct rate. In the *nouns-matching* condition, 29% of the input words can be recognized with a 97% correct rate.

Samples Used	Decision Selection Criteria								
	Overall			G02-nouns			Nouns-matching		
	M	E	C %	M	E	C %	M	E	C %
1	251	9	96	130	2	98	187	6	97
2	345	11	97	177	2	99	206	6	97
3	393	12	97	199	2	99	241	6	98
4	441	12	97	229	2	99	257	8	97
5	451	12	98	234	2	99	258	9	97
6	459	13	97	248	2	99	272	9	96
7	474	16	97	254	3	99	280	11	96
8	483	16	97	254	3	99	284	11	96
9	498	16	97	261	3	99	288	11	96
10	526	22	96	300	4	99	296	12	96

Table 8.1: Word selection performance on the original 885 neighborhoods (with 87% correct at the top choice). M=number of matches. E=number of errors. C=correct matches in percentage.

8.4 Logical Linking

An important step in logical linking is to detect the locations in the recognized text where linkings to other document elements are needed. There are two types of references: explicit reference and implicit (contextual) reference. The explicit references include the figure and table callouts, chapter and section references, and footnote and bibliography indices. The contextual references include keywords, key phrases and the domain of discourse of a text paragraph.

The explicit references are located by using graphical cues from the recognized text. The graphical cues provided by font change, case, point size, and underlines are used to locate the possible references. Figure and table callouts are indicated by upper cased words. Parenthesized text often contains figure and table callouts and bibliography indices. The system locates strings of characters delimited by an open and closed parenthesis. The objective is to to find text that has a high probability of being a reference. Footnote indices can be detected by

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

the sudden change in point size and character baseline position of the characters at the end of a word.

The implicit references are located based on the content of caption of the figures and tables and the data entries of the table. Keywords or key phrases are extracted from these text areas and are used to find text where these keys are found.

8.5 Conclusions

Our work concerns three major processes which help to build a DL database for IR. Several components of the system are shown to be useful for creating a DL database. The design of these processes stresses the importance of robustness and ease of adaptation to the processing of different documents. Output generated by these processes facilitates the IR mechanism to produce intelligent response to user queries. An adaptive approach to document understanding was presented in this chapter. Its robustness was shown to be crucial to the success in processing varied library documents. This chapter also presented an adaptation of the vector space model for information retrieval to improving the performance of a word recognition algorithm. The neighborhoods of visually similar words determined by word recognition are matched to a database of documents and a subset of documents with topics that are similar to those of the input image are determined.

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.

Nabil R. Adam Bharat K. Bhargava
Yelena Yesha (Eds.)

Digital Libraries

Current Issues

Digital Libraries Workshop DL '94
Newark, NJ, USA, May 19-20, 1994
Selected Papers

Springer

Digital Libraries Workshop, Newark, NJ, May 19-20, 1994, 119-128.
Lecture Notes in Computer Science, Springer-Verlag.