

Data Structures and Tools for Document Database Generation: An Experimental System

Rolf Bippus, Volker Märgner

Institut für Nachrichtentechnik, Technische Universität Braunschweig
Schleinitzstr. 22, D-38092 Braunschweig, Germany, e-mail: bippus@ifn.ing.tu-bs.de

Abstract. This paper is a contribution to the discussion of the structure and the elements of databases for document analysis tasks and the tools needed for database creation. It is pointed out that it is desirable to have a uniform document database that allows access to different kinds of data for different sub tasks within a complete document analysis system. Conceptual ideas pertaining to the data structure are discussed on the assumption of a hierarchical document structure. A description of an implemented data structure is also included that may serve as a starting point for further investigation and discussion. Finally, we present INSEGD, an experimental system for interactive segmentation and labelling of arbitrary documents, which is still under development along with a tool box for automatically and semi-automatically generating segmentations for support in data generation.

Introduction

The situation in document analysis today results in a vast variety of documents to be examined, together with a stronger need for division of the document analysis and recognition process into different sub tasks.

Thus the need for data acquisition on very different levels of document analysis arises. For each sub task appropriate data is required, serving as input for learning, testing and benchmarking. In this context especially the arising need for ground truth on the document segmentation level may be mentioned, as evaluation at this level becomes increasingly necessary (-> [RANDRIAMASY 94]). At the same time all statistically based recognition methods used recently like HMM and Artificial Neural Networks need very large databases for training.

We feel that the data used for developing solutions to different sub tasks of a document analysis system may not be seen independently but ought to result from the same document database for a given document analysis task. Only this guarantees a sufficient comparability between different approaches, even if they differ fundamentally and assures a good interaction of the different components.

Most large databases currently available (for example for OCR or handwriting recognition, -> [HULL 93]) only provide data for specific sub tasks and due to their fixed structure do not allow for extension to other sub tasks. What is even more important is that they make explicit assumptions about the processing stages up to the point where the data is used in a complete system thus discouraging solutions that require different kind of input.

This was the primary motivation for us to start working on data structures (memory- and file representations) for the organisation of such a database and on tools for generating and handling the database.

Many important issues on the necessities of a data structure to be used in page-readers in general are discussed in [BAIRD 94]. In [PHILLIPS 93] the organisation of a document database of machine-printed documents is described, that may serve as a database for a wider range of different tasks.

We understand our paper as a contribution to the discussion on the structure and the elements of a uniform database for document analysis tasks and the tools needed for database creation. Section 1 discusses the conceptual ideas pertaining to the data structure and the underlying hierarchical document structure. A description of the implemented structure so far is given in section 2. Section 3 presents INSEGD, a program for interactive segmentation and labelling of arbitrary documents, which is still under development along with a tool box to automatically and semi-automatically generate segmentations for support in data generation. The paper closes with an outlook on our future work.

1. Conceptual Considerations

The data structure chosen for a database as outlined above should mainly satisfy the following conditions:

- It should easily facilitate generation of input data to all stages of a document analysis system.
- It should thus reflect the physical document hierarchy as well as its logical structure.

- It must allow for the storage of all necessary information.
- It should allow for extensions in two ways: Extension of the data structure itself as well as extension of a database to previously unconsidered tasks.

The starting point of our considerations was a hierarchical document structure, the division of the document into regions of different types that recursively enclose smaller regions until basic regions (objects) are reached. On each level of the hierarchy regions may be assigned to logical "classes", for instance on the top level the document may be divided into text and non-text blocks, the non-text blocks being either images or graphical drawings.

Assuming that most sub tasks of document analysis are established somewhere within this structure, different necessities result for access to the database:

- top-down access to regions and sub regions belonging to them.

Such data will be needed among others as input and for ground truth for segmentation algorithms;

- bottom-up access, that is, access to objects of a certain "class" and their grouping into regions on higher levels, for example as an input to testing bottom-up approaches in segmentation and analysis;
- non-hierarchical, "class"-specific access to all objects of a certain class along with ground truth (labels) as an input to classification algorithms on all levels of the document structure.

Thus we have implemented a data structure that reflects these needs. Figure 1 exemplarily shows the principal structure implemented so far.

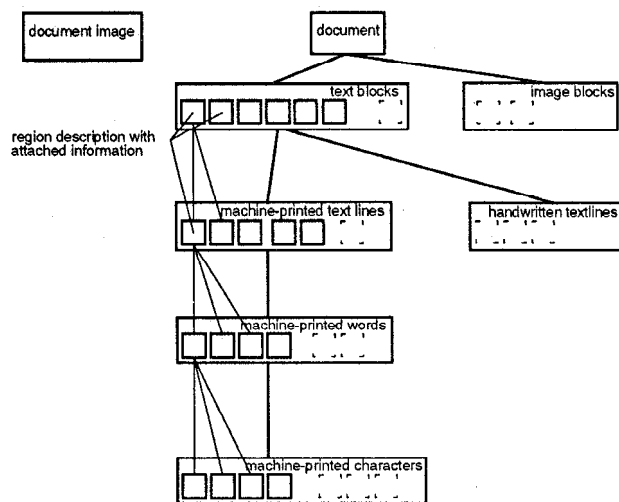


Figure 1
principal data structure for database organisation

The main feature of this structure is the fact that it models the document hierarchy on two different levels as outlined above. On the one hand it contains the physical document hierarchy as shown by the thin lines connecting parent regions with corresponding child regions that are enclosed within them. These connections are necessary to establish the hierarchical access to the database. On the other hand it models the logical structure of the type of document by gathering regions belonging to the same "class" in larger structures as indicated by the large boxes and their connections. An arbitrary number of classes is allowed at each level of the hierarchy, accounting for different alternative segmentations that might become necessary.

The iconic data is stored separately, the regions are geometrically described by their borders. This assures independence of the data structure from the type of image data used (binary, grey-level, colour ...).

2. Realisation of data structure

The realisation of the above structure is quite straightforward using 7-bit ASCII files for storing the information on the regions (objects), where regions belonging to one class are gathered in one file (->region files). The only restriction made is that the regions contained in one file may not have parent regions from different files, this being no severe restriction due to the tree structure of the document hierarchy.

Another file, also 7-bit ASCII, one for each document, contains information on the abstract document hierarchy, that is, it contains the file names of all region files along with additional information, especially on their position within the document hierarchy (->base files). The base file may also contain general information on the document.

Above all an image file containing the iconic data is supplied.

Image Files

We currently use TIFF format files.

Base Files

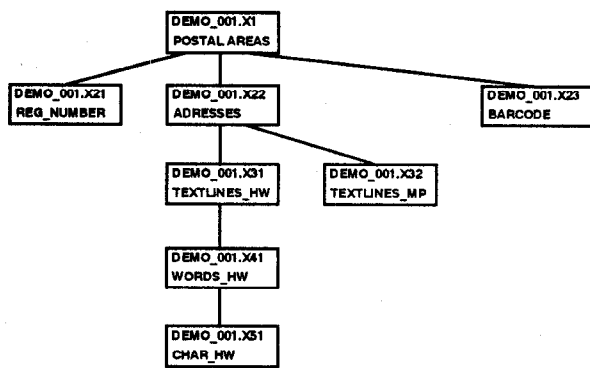
Figure 2a shows an example of a base-file, the name of the image file in the first line followed by the names and descriptions of the region files pertaining to this document. In figure 2b the corresponding abstract document hierarchy that results from the base-file is shown

```

DEMO_001.TIF
DEMO_001.X1 POSTAL_AREAS
//MAIN POSTAL AREAS
  DEMO_001.X21 REG_NUMBER
  //POSTAL REGISTRATION NUMBERS
  DEMO_001.X22 ADDRESSES
  //ADDRESS AREAS
    DEMO_001.X31 TEXTLINES_HW
    //HANDWRITTEN TEXTLINES
      DEMO_001.X41 WORDS_HW
      // HANDWRITTEN WORDS
        DEMO_001.X51 CHARACTER_HW
        //HANDWRITTEN CHARACTERS
    DEMO_001.X32 TEXTLINES_MP
    // MACHINE PRINTED TEXTLINES
  DEMO_001.X23 BARCODE
  // BARCODES

```

a)



b)

Figure 2
Example of base-file (a) along with the according
abstract document hierarchy (b)

Region Files

Each region file contains the information on all regions belonging to one class of regions in the logical document hierarchy:

- a geometrical description of the borders of the regions .
- a unique identifier
- the identifier of the parent region one level above (the link to the appropriate file being established in the base-file)
- additional information like labels, ground truth etc.

The most widely used zone descriptions for documents, up to now, are rectangles and polygons, especially because they are easy to use and have quite a compact representation.

Especially for grey level images more precise descriptions become necessary, so we added two further

possible descriptions to the above, this being a Freeman-coded description of the contour of a region and a description by means of horizontal runs. Both can be viewed as defining an arbitrary mask (bilevel image) describing the region (->[BAIRD 94]).

The format for the region files was realised following the format specified in the SAM-Project ⁽¹⁾, which is line-oriented with a token of three capital letters at the beginning of each line defining its contents. This format is a byte-stream, easy to read and OS-independent.

Due to the format of the region files any additional information that might become necessary in the future can be added by just adding an appropriate token to the list of possible tokens, without effecting the format in principal, as unknown tokens are simply ignored by the corresponding reader. Such information might be ground truth, zone type, reading direction, font type etc. (->[PHILLIPS 93]).

Figure 3 shows a commented example of such a region file with descriptions of rectangles.

```

BHE:                               /* Begin of Header */
FTY: REC                            /* file type */
PIC: DEMO_001.TIF                   /* name of image file */
WNU: 3                              /* number of regions */
DES: MAIN POSTAL AREAS              /* region type */
PFN: DEMO_001.X1                    /* Name of Parentfile */
EHE:                               /* End of Header */
BDA:                               /* Begin of Data */
.
BDR:                               /* Beg. of Data-Record */
OID: 3                              /* unique ID of region */
POI: 0                              /* ID of parent region */
OBB: REC 0 0 163 168               /* geom. zone-descript.*/
OBE:                               /* End of description */
LBL: RECIPIENT_ADDRESS              /* Label of region */
.                                  /* additional inform. */
EDR:                               /* End of Data-Record */
.
EDA:                               /* End of Data */
EOF:                               /* End of File */

```

Figure 3

Example of region-file (DEMO_001.X22 in figure 2.b)

3 Database Creation Tools

The collection of real live data presents the most laborious part in the creation of a database. In order to accomplish this task both reliably and in a sufficiently short time tools are needed to support this process in a semi-automatic way, making propositions to the human operator.

In order to explore possibilities for automatic support in a manual segmentation and labelling process, an

¹ Esprit Project 1541/2589 (SAM) Multi-Lingual Speech Input/Output Assessment, Methodology and Standardisation

experimental interactive document segmentation program (INSEGD-Interactive SEGmentation of Documents) was developed using MS Windows as a platform. We mainly focused on segmentation on all levels of the described document hierarchy rather than on generating ground truth and labelling on lower levels (text, characters etc.).

In principal the designed system supports all kind of image data (bilevel, grey level, RGB colour) although the designed tools for automatic and semi-automatic segmentation so far support operations on bilevel images in the first place. This is due to the fact that automatic segmentation algorithms on grey level images are by far more complicated and produce less reliable results, as there are no predefined borders to be used like in the bilevel case. Another restriction applying so far is a strictly hierarchical segmentation process, that allows subdivision of existing parent regions into smaller child regions.

Apart from the manual drawing of rectangles and closed polygons the following procedures for semi-automatic region generation are incorporated so far:

- shrinking rectangles to the bounding box of all black components within a previously marked region (for bilevel images only);
- automatic generation of identical equidistantly spaced rectangles, especially useful in text documents with either constant character or constant line spacing;
- generation of a polygon around previously selected black components that excludes all other black components (for bilevel images only);
- connected component analysis, generating either the outer contour of a selected black component or all contours of all black components within a selected region (for bilevel images only);
- interactively cutting an existing region into smaller child regions by arbitrary polygons.

All the above procedures work independently of the type of document to be segmented are primarily aimed at allowing a fast, manual segmentation procedure.

At the same time, the system allows the binding of existing region files, like they could be produced by specific segmentation algorithms for specific document tasks. Thus an interface exists to incorporate any desired segmentation algorithms.

4 Discussion and Future Work

We believe the presented data structure to be a valuable contribution to the discussion on what a uniform document database should facilitate, as it reflects the document hierarchy without making any further assumptions on the kind of documents within the database, like being mainly text documents.

In addition a database based on the presented data structure is open to extensions of the logical document structure. That is new data in the form of alternative segmentations and ground truth (additional region-files) can be added at any level within the existing document hierarchy becoming part of the database. This property enables the researcher to extend existing databases to meet the needs of new approaches and algorithms reusing a great deal of the information gathered before.

This implies that database creation tools must become part of such a database and must be easy to use. The tools presented for supporting the laborious task of manual document segmentation are by far not complete. Especially three missing aspects have to be mentioned. In addition to the strict top-down segmentation process, the system will have to facilitate an interactive bottom-up segmentation process as well, that is the grouping of existing regions into larger regions on higher levels, for instance starting off with a full connected component analysis. Secondly support for segmentation of grey level image data will have to be supplied, as this becomes of growing importance as the complexity of the documents to process increases. And last not least, the validation and correction procedures will have to be improved, so that the results of existing segmentation and classification algorithms for specific tasks (like text/non-text segmentation, text line segmentation) can be more easily validated and corrected by a human operator.

A last very important issue of our future work will be the exploration of special requirements for databases and data structures with respect to evaluation on the segmentation level of document analysis.

References

- [BAIRD 94] Baird H.S., Ittner D.J.: **Data Structures for Page Readers**. DAS '94, International Association for Pattern Recognition, Workshop on Document Analysis Systems, Kaiserslautern, Germany 1994
- [HULL 93] Hull J.J., Fenrich R.K.: **Large Database Organization for Document Images**. Fundamentals in Handwriting Recognition, Proc. of the NATO Advanced Study Institute on Fundamentals in Handwriting Recognition, Chateau de Bonas, France, 1993
- [PHILLIPS 93] Phillips I.T., Chuen S., Haralick M.: **CD-ROM Document Database Standard**. Proc. ICDAR'93, International Conference on Document Analysis and Recognition, Tsukuba Science City, Japan, 1993, pp. 478 - 483
- [RANDRIAMASY 94] Randriamasy S., Vincent L.: **A Region-Based System for the Automatic Evaluation of Page Segmentation Algorithms**. DAS '94, International Association for Pattern Recognition, Workshop on Document Analysis Systems, Kaiserslautern, Germany 1994