English Document Database
Design and Implementation Methodology

I. T. Phillips¹, S. Chen², J. Ha², and R. M. Haralick²

¹ Department of Computer Science
Seattle University
Seattle, Washington 98122

² Department of Electrical Engineering
University of Washington
Seattle, Washington 98195

Abstract

Systems that do OCR or any aspect of Document Image Understanding must work nearly perfectly over a broad range of document conditions and types in order to be really useful. To develop algorithms for OCR or to develop algorithms for Document Image Understanding requires that the developer have a suitable database of documents which are accurately ground-truthed so that the free parameters of the algorithms can be estimated. Customers of OCR or Document Image Understanding systems likewise must have a suitable database in order that they may accurately evaluate vendor proposed systems. Database requirements for both the developer and the customer are nearly identical. Therefore, in order to help both developer and customer, there must be the creation of a comprehensive series of databases, each specialized to a given subset of document types, or intended as additions to already created databases. The paper presents the design of a comprehensive standard document database for machine-printed documents. Our effort to produce a carefully ground-truthed document database to be issued on a CD-ROM is described in detail. The databases can be utilized by the OCR and document understanding community as a common platform to develop, test and evaluate their algorithms.

Keyword: CD-ROM, Document Database, Document Understanding, OCR
1 Introduction

The paper presents the detailed design and the implementation methodology of the English document database. Our effort to produce carefully ground-truthed document databases to be issued on CD-ROMs is described in detail. The databases can be utilized by the OCR and document understanding community as a common platform to develop, test and evaluate their algorithms. The methodology we use for the construction of the database, although costly, insures the accuracy and the integrity of the database. Accuracy and integrity are key elements that a test data set must possess in order for it to serve its purpose. Since such databases are created manually by a team of nontechnical workers, it is important to share the protocols and methodology so that others who produce such databases can leverage off our experience.

2 Motivation

Meeting performance requirements is important for Optical Character Recognition (OCR) and document understanding systems. Given an input with no noise, the ideal OCR algorithm or ideal document layout and segmentation algorithms should be able to produce a perfect output. However, if there is a random perturbation affecting the input data to the algorithm, there will be deviations from the ideal in the output of the algorithm. Given the characteristics of the perturbation on the ideal input, performance characterization attempts to establish a correspondence between the random perturbations on the input and the output of the algorithm. Performance characterization also involves the establishment of a criterion function that measures the deviation of the output under noisy conditions from the ideal output. Once the correspondence and the criterion functions are established, it then becomes possible to estimate performance of the algorithms by a computational process as done in system engineering rather than a trial and error process as is typical in many image analysis investigations.

Throughout the history of OCR research, and document layout and segmentation researchers, there has been a need for a common dataset on which to compare the performance of the algorithms. Some efforts have been made by researchers in the OCR community to make the datasets on which their algorithms have been tested available to the research community. Unfortunately, many researchers remain unwilling to do the same. Thus it becomes impossible for a researcher to verify published results by the process of replicating the algorithm or to compare the performance of a competing algorithm that she/he is developing. Worst of all, the datasets are often tuned to the algorithm that the researcher supplying the datasets has developed. Thus the researcher obtaining the datasets has no alternative but to tune her/his algorithm to the dataset and this does not promote good research. It is time for comprehensive standard document databases to be constructed and made them available to researchers. Such databases would serve to provide uniform platforms on which researchers could compare the recognition accuracy of their OCR algorithms. Such databases need to be developed for documents in each one of the world’s major language, such as Roman, Hebrew,
Arabic and Farsi, Kanji, Hangul, Devnagiri and Cyrillic.

It is also necessary to develop document degradation models to provide researchers with a mechanism for introducing random perturbations on noise-free ideal images. These degradation models will be developed based on the kinds of degradation found in real-life photocopying and fax transmission, ink bleeding and page aging, as well as others. The document degradation software will simulate these real-life degradations. It can be used to generate controlled degraded document pages for OCR algorithm development, as well as for performance evaluation and testing of the algorithms. Such synthetically degraded images give unlimited extension to the real data sets in the database without having to provide additional ground truth for the generated documents.

3 English Document Database Overview

In order to be useful for developers of OCR algorithms, an English language database that should reflect the full range of machine-printed documents. The database should at least include the following document page types of documents:

- Articles: journals, conference proceedings, books, dictionaries, etc..
- Business letters: letters and memorandums of all kinds.
- News papers/magazines: All sizes and colors.
- Maps: Street maps, terran maps, etc..
- Forms:
- Manuscripts:
- Engineering Drawings: including CAD/CAM.
- Advertisements

For each type of document, a variety of documents of various formats and quality will be presented. These documents will be drawn according to the frequency of their usage. The database will be packaged and distributed on an ISO 9660 compatible CD-ROM in order for it to be accessible from a wide variety of platforms. The Rockridge protocols extensions to ISO 9660 will be used to incorporate UNIX file system information into the CD-ROM due to the wide acceptance of the UNIX operating system in the OCR research community. The use of the Rockridge extensions does not in any way compromise the ability of the CD-ROM being read by non-UNIX platforms such as PCs.

The database will include digitized scanned greyscale and binary documents, synthesized noise-free documents, photocopy and fax degraded documents (degraded through both the
real process and simulation). All binary image files on the CD-ROM will be compressed using the CCITT Group IV standard for bi-level facsimile image compression. The database will include the page attributes and the zoning information of each document page, as well as the ground truth for each document. For text zones, the ground truth data will be in two forms - 1) an ASCII character sequence (zone-based), or 2) individual characters with their position on the page (page-based.) For mathematical zones, the form of ground truth may be developed and provided. For line-art, halftone zones, etc., there will be no ground truth.

Also, the CD-ROM will contain document degradation software that simulates real-life photocopying and fax communication channel process. The degradation software can be used to generate additional degraded document pages for the performance evaluations and testing of OCR algorithms and document layout and segmentation algorithm. The software that evaluates the performance of OCR algorithms will be provided within the database.

4 Environment

End Users

The end user of the document database would be OCR researchers and document layout and segmentation algorithm developers. The contents of the database would be used to test and benchmark the developed algorithms and their sub-components.

Hardware and Software Requirements

The CD-ROM can be accessed by any system that has a CD-ROM drive with software drivers that can access ISO 9660 compliant CD-ROMs. Sufficient disk space will also be necessary to download images from the database. Alternatively, when degraded image documents are created by software on the CD-ROM, sufficient disk space will be needed to write the images onto disk.

Programming Language

The programs residing in the CD-ROM will be written in the C programming language. In order to compile and use the programs (UNIX Makefiles will be provided), the platform to which the CD-ROM drive is attached must have a C language compiler/interpreter.

5 CD-ROM Contents and Organization Overview

The CD-ROM has two logical compartments: the software compartment and the document page compartment. The software compartment contains the data compression and decompression software, the OCR performance evaluation software, and the 'zone box' position
and size evaluation software, the photocopy degradation software, and the fax degradation software. The descriptions of this compartment is given in section 5.1.

The document compartment contains all document pages, page processing facts of the documents (digitization parameters, noise degradation model and parameters, source, etc.), and the page attribute record files that describe various attributes of the document pages. The compartment will also contains the zoning information of each document page and ground truth for text zones on the page. The descriptions of this compartment is given in section 5.2.

Since the document database will be packaged on a CD-ROM. The file names and directory structures will be in compliance with ISO 9660. As an option, we will use Young Mind's Original Name Recovery protocol to permit recovery of original UNIX filename and file system information. The general file name conventions for the document database are given in section 5.3.

5.1 Software Compartment Contents

Data Compression and Decompression Software

All the scanned and simulated documents in bitmap (TIFF) format on the CD-ROM are supplied in compressed form. The compression algorithm used is the CCITT Group IV bi-level image compression standard. The user can then use the decompression program provided in the CD-ROM to uncompress the compressed data files.

OCR Performance Evaluation Software

OCR performance evaluation software will be developed and provided. Given a list of document zone IDs and OCR outputs of the zones, the algorithm will evaluate the output of OCR algorithm against the corresponding ground truth residing in the CD-ROM. A set of contingency tables for characters and mis-recognized words will be computed and output by the algorithm. The requirement of the software package is given in [2].

Photocopy Degradation Software

Software is also provided that simulates two selected document degradation models. One is Henry Baird's model [6] and the other is currently developed by researchers at the Intelligent Systems Laboratory. Given a document file (binary image file) and degradation model parameters, the user can run the photocopy degradation program to degrade the document as desired. The requirement specification of the software is given in [3].

FAX Communication Channel Degradation Software

A software package for simulating the degradation of printed documents when transmitted using the CCITT Group III fax transmission standard will be implemented. The following four models will be used in the software package.
- Pareto Model
- Fano Model
- McCullough Model
- Fritchman Model

The descriptions of the above four models are given in [4].

5.2 Document Page Compartment Contents

Document Page Files

The document page compartment includes a set of document image files. Each document image corresponds to one document page. The document images can be classified into the following categories:

- Scanned/digitized grey scale images of real documents.
- Scanned/digitized binary images of real documents.
- Synthesized noise-free binary TIFF formatted image.
- Synthetic degraded binary TIFF formatted image.

The real document pages will include a set of document which are taken from the set of synthesize noise-free documents and degraded through real processes - both by successively photocopying or fax transmission. The degraded document pages will also include a set of synthetic degraded documents (for convenience) that are degraded by the same degradation software that will be provided in the database. The source document of the degraded pages will come from a set of selected pages from category 2 and 3.

Scan Digitization Parameter Files

For the scanned document pages in the database, the parameters that control the scanning process of the documents will be stored as a set of ASCII files. The definitions of the scan/digitization parameters are given in Appendix.

Degradation Model Parameter Files

For a synthetically (either photocopy or fax transmission) degraded document page, a set of parameters that control the degrading process of the document will be provided. The definitions of the record fields for these two records are given in Appendix.
Page Attributes

For each document page in the database, there is a set of descriptive attributes which describe the various attributes of the page. Each document page type (journal, letter/memo, etc.) has its own set of attributes. For technical journals/reports, the attributes include page condition, page bounding boxes, page contents, page layout, font information, publication information, etc. For letters and memos, a subset of the journal page attributes are used. The journal page attributes definitions are given in Appendix.

The advantage of defining a set of document page attribute records for each document type over a set of general page attributes for all document types is that it allows one to add another document type to the database without any change to the database design.

Zone Attribute Files

Each document page will be zoned manually according our zoning conventions [5]. Each zone in a page is associated with a set of zone attributes that describes the contents of the zone. We define a distinct zone attribute record for each document type (journal, etc.). The definitions of the zone attribute are given in Appendix.

Ground Truth Files

The database provides the ground truth for all text zones on all document pages. (For mathematical zones, the form of ground truth may be developed and provided. For line-art, halftone zones, etc., there will be no ground truth.) The format of the ground truth is given as character sequences: the correct character sequences (with line breaks between sequences) within the zone.

In addition, each LaTeX generated document page resides in the database, we provide a ground truth file that contains character positions of all characters on the page. The format of the ground truth is given in a character position sequence: the sequence of characters with the position (coordinates) and the size of each character in the zone. (This format only for synthesized and degraded document pages arising from the LaTeX generated documents in the database.

The ground truth files will be used by the OCR evaluation software reside in the database. The software evaluates the performance of OCR algorithms.

All non-ASCII symbols will be represented in LaTeX-alike syntax. The substitution table will be provided.

Page and Zone Bounding Annuluses Information Files

The database also provides an annulus box for each page header, page footer, page live matter and each zone on all scanned document pages. (The live matter of a document page is the usable area of the page between the margins [1], pp4-9.) We define an annulus as the area between two rectangular boxes (one enclosing the other.) An annulus will be represented as the two rectangular bounding boxes (the 'inner' and the 'outer') that encloses
an area of a document page. The area can be the header area, the footer area, or the live matter area of a document page, or a zone of a document page. The inner bounding box will approximate the smallest rectangular box that contain the entire area content and nothing else. The outer bounding box will approximate the largest rectangular box that contain the entire area content and nothing else. The definitions of the annulus information are given in Appendix.

5.3 File Name Convention

The general file name conventions for the document database are as follows:

- A legal file name will consist of at most 8 characters (26 capital English letters and 10 digit numbers) followed by a period "." and followed by a 3 character file name extensions.

- The first character of the file name must be a capital English letter.

Under our current design, the document files have additional file name constraints to make them more identifiable. For example, the first four characters of the file name represent the document page ID. The fifth character of the file name is a sequential document page copy number. There may be multiple qualities of the same document page residing in the database. The fifth character is used to differentiate them. The sixth to eighth characters of the file name are used to identify the category of the file (scanned binary, scanned grey-scale, page attribute/condition record file, zone attribute record file, ground truth, etc.) The filename extension will indicate the file format (.TIF for image file, .TXT for ASCII file and .TEX for LaTeX file).

6 Document Database Implementation Methodology

6.1 The Implementation Process

The implementation of English Document database is very complex. First of all, a journal article page to be included in the database needs to be manually selected from its source (e.g., from journals in library or from LaTeX files). The selected page must satisfy the journal page selection requirement at the time of the selection. (Done in the journal page selection process). The publisher's and copyright holder's names and addresses must be recorded and a permission letter must be made and sent to the copyright holder for permission to include the page in the database. (Done in the copyright permission request process). Recall that, each journal page in the CD-ROM is associated with an image file (binary, grey-scale, or both), a page bounding boxes file, a page content attribute file, a page condition file, a zone attribute file, and a zone-based ground truth file. The names of these files are uniquely
identifiable by the identification number (id) of the page. Figure 1 shows a version of the CD-ROM file directory structure.

To create these files, the page must first have a unique id (done in the page id assignment process). The page must have its page content attributes and its physical appearance (page condition) recorded. This is done in the page attribution process. All `legal` zones on the page must be identified and given unique zone ids (done in the zone id assignment process). Each zone on the page must have its zone attributed and have the attributes recorded. (done in the zone attribution process). Ground truth for the recorded page content attributes and the page condition attributes, the zone attributes, and zone-based ground truth are created in double, independently, by two individuals. (Done in the data entry process). The data pairs are verified and corrected (if necessary), iteratively, until the contents agree one with another. (Done in the verification process). The verification and process is also done in double (verified pairs) and independently, by two individuals who did not create the data pairs. Finally, the verified pairs are themselves being verified and corrected (if necessary), iteratively, until the contents of the verified pairs agree one with another. Only doubly created and doubly verified and corrected files are put on the CD-ROM English document database.

Figure 2 shows the top view of the process and data flow diagram of the database implementation process. All the procedures and protocols we have design for the implementation have being used through out every stage of the implementation process. Without these protocols, the implementation of the database would not be possible within the given one year schedule. The complete discription of these procedures and protocols are given in [5].

### 6.2 Document Page Selection Methodology

The methodology we use to insure that the database contains a wide variety of journal articles from all fields of science, and that it includes various page formats, page layouts, and font type sets is as follows.

- **Journal Selection Rules:**
  1. Use the first character of the journal names as the journal selection key.
  2. Select five to ten journals for each letter. (The number is proportional to the number of available journals that have the same key letter).
  3. Only one journal volume can be selected from its entire publication set.
  4. Select no more than ten journals and no more than fifty pages at one setting (a human factor).

- **Article Page Selection Rules:**
  1. Randomly select one article title page from each selected journal.
2. Randomly select one complete article from each set of five journals.

3. Select no more than five article pages from a single journal volume, except when a complete article is called for and the selected article spans more than five pages.

4. The selected pages should have attributes that are required and listed on a preassigned ‘page attribute wanted list’. The page attribute wanted list is given at the time of selection and is determined either manually or automatically based upon the attributes occurring in all the previously selected article pages.

6.3 Implementation Procedures and Protocols

Page Selection Protocol

The following procedure is for selecting a set of journal pages from a library that has a large collection of technical journals.

step 1. Have a copy of this procedure and a copy of the Journal Selection Report Form (JSR Form). (The JSR Form is given at the end of this protocol).

step 2. Complete the JSR Form with the person who assigns the task to you. When you complete the Form, you should know the number of journals you need to locate, the number of pages you need to select from each journal, and the year(s) and the initials of names of journals you selected.

step 3. Ask for a box of small paper clips, and a box of extra-large paper clips. Also have a note pad, a pen, and a few pads of 3M stick-ons.

step 4. Ask where and how to pay for the copying. Be sure to have enough change if using a library copying machine.

step 5. Go to a library that has a large collection of technical journals. (If you are on UW campus, go to the Engineering Library).

step 6. Select the required number of journals from the library. Make sure the journals you selected satisfy the requirement stated in the JSR Form.

step 7. Go to the copying location that you are instructed in step 4.

step 8. For each of the journals that you have selected do the following:

1. Locate the page that has the journal name, the publisher’s name and copy right permission address on it (Be sure the address is the one for obtaining copyright permission). Make two identical copies of the page.

2. Select the required number of pages, listed in JSR Form, from each journal. (Be sure the selected pages satisfy the attribute requirement listed in JSR Form).
3. Make two (2) identical copies of each selected page. (Please do NOT make one copy first, then making the other from that one. This is very important because we need two IDENTICAL copies).

4. Examine each copy to be sure that the journal page is properly copied—no rotation, everything on a journal page is on each of the two copies, no blurring, etc...

5. Separate the stack of copies into two sets with the journal information page on top. Clip each set with a small paper clip.

6. Clip the two sets together with an extra-large paper clip.

7. Repeat step 8 if you have more journals to do.

At this point, you should have the same number of large stacks as the number of journals you have selected.

step 9. Find a desk and have one set of the stack of the copied document pages with you. For each of the journals that you have selected record on your note pad the following: (You may be asked to enter the information directly to a computer file, instead of writing them on the pad).

1. Publisher’s Name and Address:
2. Journal’s Name:
3. Volume Number:
4. Issue Number:
5. Selected Page Number List:

step 10. Hand the following to the person who assigned this task to you:

1. The stack of copies
2. The completed JSR Forms
3. The recorded publication list (Either on the pad or a computer print out).

Journal Selection Report Form (JSR Form)

- Your Name:
- Date needed:
- Number of Journals: (10) (20) (30) (____)
- Years of the selected journal:
- Journal name initial(s): (Can circle one or more:)
  (A) (B) (C) (D) (E) (F) (G) (H) (I) (J) (K) (L) (M) (N) (O) (P) (Q) (R) (S) (T) (U) (V) (W) (X) (Y) (Z)

- Number of complete articles: (0) (1) (2) (3) (----)

- Number of single pages from each journal: (5) (----)

- Page attributes consideration:
  - Number of title pages:
  - Number of reference (last) pages:
  - Other attributes:

Protocol for Obtaining Copyright Permission

The following protocol is for obtaining copyright permission for a set of journal pages to be included in the database.

Procedure

Follow these steps:

step 1. Start with a publication list and a stack of unused envelopes.

step 2. Open the database publication information file “copyright” in the cdrom directory and add the publication list to the file. Save the file. (If the file does not exist, open a new file by that name and enter the publication list to the new file. Save the file).

step 3. Copy to the copyright template letter file “template.tex”.

step 4. For each journal on the list, do the following:
  1. Edit the copyright template file by modifying the publisher’s name, the address, and the page number list.
  2. Print the modified letter and have it signed.
  3. Get an envelope and write the name and address on it.
  4. Put the letter into the envelope and put a stamp on it.
  5. Repeat step 4 until you finish all the journals on the list.

step 5. Mail the stack of addressed envelopes.
Protocol for Document Page ID Assignment

A document page ID is a four character string. Starting with a capital letter and followed by three alphanumeric characters. The three characters make a serial number. In our numbering system, we start with number 000 and increment it serially till 009. At that point instead of going to 010, we go to 00A, 00B, up to 00Z, and we then go to 010. So our numbering sequences are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, in that sequence. As an example, “A000” is a legal document ID, while “0AAA” is not.

Protocol

step 1. Start with a set of copies of the selected document pages that are ready for ID assignment.

step 2. Ask for the starting document ID. Let this ID be the “next ID”.

step 3. Do the following steps until no more document pages are left (use a black or blue ink pen):

   1. Put the “next ID” on the upper righthand corner of the document page on the top of the document set, and then remove the page from the top.

   2. Increment “next ID” by one number. (The next four character sequence that follows the last used document ID in our numbering sequence).

   3. Repeat step 3.

Protocol for Document Page Attribute Assignment

The following procedure is for assigning page attributes for a set of document pages that have been selected and IDed. You are encouraged to write comments or suggestions on the left margin of the protocol.

Protocol

Follow these steps:

step 1. Start with a set of document pages that has been page-IDed and a copy of its page attribute definition document. If this task is new to you, study the attribute definitions in the document.

step 2. Ask who is the person responsible for verifying your filled attribute sheets when you are done.
step 3. Ask for at least as many blank page attribute sheets, and small paper clips as the number of document pages in the set. Also ask for two dark lead pencils.

step 4. Go through each document page in the set and make sure each page has ID written on it. Remove any document page that does not have an ID.

step 5. For each document page in the set do the followings:

   1. Have a blank page attribute sheet and the document page in front of you.
   2. Write the four character string, on the top of the document page, into the Document ID slot (slot 1) of the attribute sheet.
   3. Sequentially, for each attribute on the sheet, CIRCLE the value that applies. In case nothing in the value list applies to the document page, circle item (not clear) on the list, or ask for advice if the person in charge of the task is available.
   4. Examine the attribute sheet to make sure all attributes are correctly circled and nothing is left blank.
   5. Clip the attribute sheet on top of the document page.
   6. Repeat step 5 until all document pages have been attributed.

step 6. Hand the filled sheets to the person who is assigned to verify your work in step 2.

Protocol for Drawing Zone Boxes

The following procedure is for drawing zoning boxes on a set of document pages. You are encouraged to write comments or suggestions on the left margin of the protocol.

Protocol

Follow these steps:

step 1. Start with a set of document pages that need to be processed and a copy of zone attribute sheet.

step 2. Study the zone attribute sheet to understand our zoning rules. The following are some of the yes/no zone rules.

   - A zone must fit within a rectangular box.
   - The largest zone is one paragraph and the smallest zone is one character.
   - Each paragraph is a zone if it fits within a rectangle. Otherwise, paragraph can be broken into rectangular blocks.
   - A displayed math equation area (with equation number) is a zone.
   - Figures, tables, pictures, etc. are zones themselves.
• A caption of a figure or table is one zone. (If the caption is inside of a figure or a table, do not zone the caption).
• A list item is a zone.
• Each reference item within a reference list is one zone.
• Page number is one zone.
• A drop cap is one zone.
• A handwritten text blob is a zone.
• The content of a zone is on the ‘zone content’ attribute list and the zone can be labeled using one of the ‘zone label’ attributes.
• A line drawn on the page (either vertical or horizontal) is one zone.
• A zone may NOT cross text columns.
• A box drawn on the page is one zone. (A box may contain other zones. This is the only time when zones can overlap).

step 3. Ask who is the person responsible for verifying your zoned document pages when you are done.

step 4. Go through each document page in the set and make sure each page has ID written on it and the ID are in sequence. If you find any page does not have an ID or any page is missing, inform the person who assigned you the task.

step 5. For each document page do the followings:

1. Draw a rectangular box circumscribing each ‘legal’ zone on the page (Please do your best to insure that the entire zone content is within the rectangular box and do NOT write on top of any text).
2. Examine the attribute sheet to make sure all attributes are correctly circled and nothing is left blank.
3. Clip the attribute sheet on top of the document page.
4. Repeat step 5 until all document pages have been zoned.

step 6. Hand the set of zoned documents to the person who is assigned to verify your zoned sheets.

Protocol for Zone ID Assignment

A zone ID is a three character string in the following sequence: 000, 001, 002, ... ; 009, 00A, 00B, ... ; 00Z, 010, 011, 012, ... ; 019, 01A, 01B, ... (The zone ID for each page begins with 000, 001, 002, .... and so forth.)

Protocol
step 1. Start with a set of document pages that have zone boxes on them (The zone boxes on the pages should have been verified).

step 2. Ask who is the person responsible for verifying your work when you are done.

step 3. For each zoned document page do the following:

Assign a unique zone ID for each rectangular zone box on the document page. The ordering of IDs given to the zones is NOT arbitrary. The rules are as follows:

- Header area first, live matter area second, and footer area last.
- Within each area, number the zones in a Top-to-bottom and Left-to-right fashion (The ID must written next to the zone box, unambiguously).
- Repeat step 4 until done.

step 4. Examine and make sure that each zone box is given an ID, and the ordering of the zone IDs are given in accordance with the established rules.

step 5. Hand in the processed documents to the person who is assigned to verify your work (in step 3).

Protocol for Zone Attribute Assignment

The following procedure is written for assigning zone attributes for a set of document pages that have been zoned and IDed. You are encouraged to write comments or suggestions on the left margin of the protocol.

Protocol

Follow these steps:

step 1. Start with a set of document pages and their corresponding page attribute sheets and a stack of zone attribute sheets. If this task is new to you, study the definition of the zone attributes.

step 2. Ask who is the person responsible for verifying your attribute sheets when you are done.

step 3. Make sure you have at least twenty times as many blank zone attribute sheets as the number of document pages in the set. Also have two boxes of large paper clips and two black (or blue) ink pens.

step 4. For each zoned document page in the set do the followings:
1. Write the document page ID (the four character string written on the upper-right-hand corner of the document page) into the Document ID slot (slot 1) for each blank attribute sheet in the stack.

2. Fill in the zone id slots in the stack, in the following sequence: 000, 001, ... , to the last zone id on the page.

3. Sequentially, for each attribute on the sheet, CIRCLE the value that applies. In case nothing in the value list applies to the document page, circle item (not clear) on the list, or ask for advice if the person in charge of the task is available.

4. Examine the attribute sheet to make sure all attributes are correctly circled and nothing is left blank.

5. Repeat 2-6 until all zones on the document pages have been processed.

**step 5.** Put the document page on top of the zone attribute sheets and clip them together with a large clip.

**step 6.** Repeat Step 4 until no more document page to be processed.

**step 7.** Hand the stacks of attribute sheet set to the person who is assigned to verify your work (in step 2).

**Zone-based Ground Truth Data Entry Protocol**

This section describes the data entry protocol for the Zone-based ground truth. The document is written for the person who is assigned to do the data entry.

**Protocol**

For each document page in the given data entry material do the following steps.

**Step 1:** For each document page, open a new file.

The new file name will have the form "XXXX.FML", where XXXX is the four-character string (Document page ID) handwritten on the top of the document page. The four-character string should start with a capital letter then followed by 3 characters (can be a combination of capital letters and numbers). FML are your first, middle, and the last names initials. For example, if the document page ID is H123 and your initials are STP, then the file name would be H123.STP (If the file name has been used and it is in the directory you are working on, stop and consult with your contact person).

Once the file is opened and named, you are ready to enter the text within the zone box. Do one zone after another follow one of the numbering sequences:
• 000, 001, 002, 003, ..., 008, 009, 00A, 00B, 00C, 00D, 00E, ..., 00N, 00O, 00P, 00Q,... 00Z, 010, 012, ....

**Step 2:** After opening a new file,

- Enter ZZZZZ (5 Z's or more) and follows by a return.
- Enter the document page ID, then follow that by a return.
- Enter the zone ID, then follow that by a return.

For example, if the four-character string (document ID) handwritten on the top of the document page is H123, if the three-character string (zone ID) handwritten next to the zone is 008, then you should have enter the followings:

```
ZZZZZ  [RETURN]
H123   [RETURN]
008    [RETURN]
```

**Step 3** Enter zone text.

Enter the text within the zone box line by line. Use the return key at the end of each text line. When you encounter any character(symbol) that you can not find on your keyboard, please use the substitution string for that character. If you can not find the substitution string for the symbol, enter \????(5 ?’s) instead (The substitution strings for all special symbols are defined in the special symbol table in the following section). For example, if a zone box contains the following three text lines:

The following two symbols $\alpha$ and $\beta$ are not on the keyboard.
The next character is a \\.
Treat the following strings, $x_2$, $x^2$, $x_2$, $x^{a+b}$ differently.

Then you enter these three lines as:

The following two characters $\alpha$ and $\beta$ are not on the keyboard.
The next character is a \\backslash.
Treat the following strings, $x_2$, $x^{\{2\}}$, $x_{\{2\}}$, $x^{\{a+b\}}$ differently.

**Step 4:** When you finish one zone, repeat Step 2 to Step 3 for all zones on the document page.

**Step 5:** When you finish one document page, repeat Step 1 to Step 4 for all the document pages in the data entry material.

**Step 6:** If you are using an ISL SPARC station, record the directory of the data files on the Data Entry Form (at the end of this protocol). And then, return this document(along with the Data Entry Form) and the data entry material to your contact person.
If you are using an IBM or Macintosh PC for the data entry, then convert all data files to flat ASCII files, copy the ASCII files onto a floppy disk. Then deliver the floppy disk, this document (along with the Data Entry Form) and the data entry material to your contact person.

Document Page Attributes Data Entry Protocol

This section describes the data entry protocol for a set of document page attributes. The section is for the person who is assigned to do the data entry.

Protocol

The following are the steps to do your data entry.

Step 1: Open a new file. The new file name will be "PAGEATTR FML", where FML are your first, middle, and the last names initials. For example, if your initials are STP, then the file name would be PAGEATTR.STP. (If the file name has been used and it is in the current working directory (or folder), consult with your contact person.)

Once the file is opened and named, you are ready to enter the information from the page attribute sheets.

Step 2: For each sheet in the stack do the followings:

1. Type "PPPPP" (5 P's) followed by a return key.

   PPPPP [RETURN]

2. For item 1: Enter the number 1, a space, the document page ID and then press the return key. For example, if item 1 appear as

   1. Document Page ID: ___H0A9____, then enter the following:

   1 H0A9 [RETURN]

3. Enter all the other items on the Page attribute sheet in the following manner:

   - Enter the item number, a space, the selected value, and then press the return key. (one item per line).

   - If the item is a selection among different options, enter the name of the circled (or underlined) option and then press the return key. For an example, Example 1, if item 2 appears as

   2. Type of Document: (journal) (report) (letter) (memo)

   If (journal) were circled (or underlined), then enter:

   2 journal [RETURN]

   Example 2, if item 9 appears as
9. Full article Present: (yes) (no)
If (no) was circled (or underlined), then enter:
   9 no [RETURN]
   • If no option is circled, type the written comment on the space after "specify other ________________".
   • If nothing is circled (or underlined) and nothing is written on the space after "specify other ________________", then enter "?????" (5 ?’s).
   • If the item has a line next to it, then enter the written text (if any). For an example, if item 4 appear as:
     4. Journal Name: ___Applied Mathematics and Computation___
     then enter the following:
     4 Applied Mathematics and Computation [RETURN]
   • If the item has a line next to it, but there isn’t any written text, enter "?????" (without quotation mark). For an example, if item 8 appear as:
     8. Page Number: ________________
     Then enter the following:
     8 ???? [RETURN]

Step 3: Repeat Step 2 for all sheets in the stack.

Step 4: When you finished all sheets within the stack, close and save the file.

Data Entry Form

1. Data Entry person
   • Name:
   • Initials:
   • Phone:
   • e-mail address (if any):

2. Contact person
   • Name:
   • Phone:
   • e-mail address (if any):
3. Check the type of computer you will use:
   - an ISL Sun SPARC station
   - an IBM PC (or compatible)
   - a Macintosh PC

4. Data Entry File Directory (only if you use an ISL computer):

**Document Zone Attributes Data Entry Protocol**

This section describes the data entry protocol for the zone attributes. The document is written for the person who is assigned to do the data entry.

**Protocol**

The following are the steps for your data entry. For each stack of sheets in the given data entry material do the following:

**Step 1:** Open a new file. The new file name will have the form "XXX.XFML", where XXXX is the four-character string written on item 1 (Document page ID) of the zone attribute sheet. The four-character string should start with a capital letter followed by 3 characters (can be a combination of capital letters and numbers). FML are your first, middle, and the last names initials. For example, if the document page ID on item 1 is H123 and your initials are STP, then the file name would be H123.STP. (If the file name has been used and it is in the current working directory, stop and consult with your contact person).

Once the file is opened and named, you are ready to enter the information from the zone attribute sheets.

**Step 2:** For Each sheet in the stack do the followings:

1. Start by typing "ZZZZZ" (5 Z's) followed by a return key.
   
   
   ZZZZZ [RETURN]

2. For item 1: Enter the number 1, a space, the document page ID, and then press the return key. For example, if item 1 appears as
   
   1. Document Page ID: ____H123____

   then enter the following:

   1 H123 [RETURN]
3. For item 2: Enter the number 2, a space, the zone ID, and then follow that by a return. For example, if item 2 appear as

2. Zone ID: ___09___
then enter the following:

2 09 [RETURN]

4. Enter all the other items on the zone attribute sheet in the following manner:

- Enter the item number, a space, the selected value, then a return. (one item per line).
- If the item is a selection among different options, enter the name of the circled (or underlined) option and follow it by a return. For an example, For example, if item 6 appears as

6. Zone Orientation: (non-text none) (vertical) (horizontal)
and if (vertical) were circled (or underlined), then enter:

6 vertical [RETURN]

- If no option is circled, type the written comment on the space after “specify other ___________________”.
- If nothing is circled (or underlined) and nothing is written on the space after “specify other ___________________”, then enter “?????” (5 ?’s).
- If the item has a line next to it, then enter the written text (if any). For an example, if item 8 appears as:

8. Dominant Font Size : ___12___ points
Then enter the following:

8 12 [RETURN]

- If the item has a line next to it, but there isn’t any written text, enter "?????" (without quotation mark). For an example, if item 8 appears as:

8. Dominant Font Size : __________ points
Then enter the following:

8 ???? [RETURN]

Step 3: Repeat Step 2 for all sheets in the stack.

Step 4: When you have finished all sheets within the stack, close and save the file.

Step 5: Repeat Step 1 to Step 4 for the next stack (if any). If you are using an ISL SPARC station, record the directory of the data files on the Data Entry Form (at the end of this protocol). Then, return this document (along with the Data Entry Form) and the data entry material to your contact person.

If you are using an IBM or Macintosh PC for the data entry, then convert all data file to flat ASCII files, copy the ASCII files onto a floppy disk. Then deliver the floppy
disk, this document (along with the Data Entry Form), and the data entry material to your contact person.

Data Entry Verification/Correction

Preliminaries

Under our design for the implementation of the database [7], the page attributes, the zone attributes, and the text zone ground truth of all scanned document pages in the CD-ROM will be typed (accordance with the data entry protocols) into computer files by two independent data entry personnel. We have adopted the double data entry model for the creation of the database. Although costly, this insures a high accuracy of the database.

The pairs of data entry files (typed by two different people using identical data entry material) are checked for their data entry accuracy by using the UNIX diff command on those files. If the diff command finds differences between a pair of files, the person doing the verification will make corrections to both files and iterate the procedure until diff finds no difference in the corrected files.

The verification and correction procedure will also be done by two independent verifiers who did not do the data entry. A similar verification (using diff) and correction procedure also apply to the pair of files come from the two independent verifiers until all pairs agree to one another.

This document contains the protocols for the verification and correction on a set of paired data entry files (the zone attributes, the page attributes, and the ground truth). This document is written for the person who is assigned to do the verification and correction. Therefore, when we say you in this document, it is directed to the person who will do the verification and the correction.

Assumptions

If you do not have an account on our computer system, you will need to get our system manager to set up an account for you because you need an account on our system to do the work.

We assume that you are familiar with the SUN multi-window environment and know how to use one of the two editors: vi or emacs. And we assume that you are familiar with the UNIX diff command and the meaning of the diff’s outputs (If you don’t know yet, please use UNIX man diff and study the syntax and semantics of diff command).

You may assume that all files in the same directory (within the database) contain the same kinds (page attribute, etc.) of information. And all file names have the form of DDDD.PP or DDDD.PPP where DDDD is a four character document page ID and PP (or PPP) are the initials of the typists who did the data entry. Thus, each pair of files that are typed by two independent typists, based on the same data entry material, will have the identical document page ID and distinct initials. For example, if the two typists initials are
STP and PLP, then the pair of the zone attribute files associate with the document page, A001, typed by the two typists will be A001.PLL and A001.STP. These two files will be in the directory DataEntry/ZoneAttr/unverify/.

Protocol: Data Entry Verification/Correction

This section contains the protocol for the verification and correction on a group of pairs of files (typed by two different people based on the same data entry material). Some pairs of files may be identical but not necessarily all of them. When there are differences, at least one file is in error. To determine the difference between pairs of files and make the corrections, follow the steps below:

Step 1: (pre-steps) Do the following.

1. Ask for the complete path of the directory (source directory) where the files reside.
2. Ask for the hard copies of the data entry material which the files were based on.
3. Create a working directory, say, “work”, and change your current directory to “work”.
4. Use the UNIX “cp” command to copy all the files you need to work on from the source directory to your current directory, work, as follow:
   `cp source/* .`
5. Then, do ‘ls * | lpr -Plw’ to print out and see the naming pattern of your files.
6. Have at least three windows on the monitor of the SUN to work with. Let’s call these windows as W1, W2, and W3. Arrange the positions of the three windows to your liking on your monitor.

Step 2: (verification steps) Have the file names list and the hard copy data entry material in hand. Do the verification and correction on all pairs of files in the order of their names on the file names list. From the first pair to the last pair in the file names list do the following:

1. Use window, W3, and type: `diff -iw file1 file2 | more`
   (file1 and file2 are names of the file pairs. Please get familiar with `diff` command.)
2. If the return from `diff` is not empty, you will see on the display W3 the set of differences between the two files. For each different instance, `diff` gives the line numbers of the two files and the contents of the difference of the two files (using < for file1 contents and > for file2 contents). If the return from `diff` is hard to see on the display window W3, do `diff -iw file1 file2 | lpr -Plw2` to print out a hard copy.
3. use emacs (or vi) on window W1 to open file1 and window W2 for file2.
4. Locate the error line number in the file2 and in the file2. Use the hard copy data entry material as the ground truth and correct the error file(s) for all difference that `diff` found. (Modifications should be done on both files).

5. When you believe that you have corrected all errors that `diff` found, go to window W3 and repeat the `diff` command line (use `!!`). If `diff`’s return is not empty, repeat the last three step until `diff` return empty. (At this point, the two files are the same). Remark: If you wonder why don’t we ask you to correct the first file and then copy the corrected file to the second file. It will save time (and money) and have the same result, right? Wrong! We believe that making the corrections on both files will result in a higher final data accuracy. This is very important to us.

Repeat the modification steps until all files are verified and corrected.

**Step3:** *(file renaming steps)* Do the following.

1. Ask for the program that will change the file names from an old name to a new name.
2. Ask how you should rename your corrected files (The new file name convention).
3. Run renaming program to change all file names to their new names.
4. Send an e-mail to the person who assign the task to you about the file directory and the new names of the files (The corrected files will be transferred to its destinate directory for further verification).
Figure 1: A version of the CD-ROM file directory structure
Figure 2: The top view of CD-ROM database implementation process
Reference


Appendix

This appendix gives the definitions of all records that constitute record files within the document page compartment of the CD-ROM.

Scan Digitization Record

For the scanned document page, the set of parameters that control the scanning process of each document will be recorded into the document database in the form of records. To access the parameters, just access the fields of the record. The scan digitization record has the following fields:

Record Field Definitions:

- Document ID:
  A unique four character string, the four character string (id) assigned to the document page. The document ID can also be thought of as a filename that uniquely identifies each document in the database.

- Machine Type :
  The type and made of manufacture (ex. Fujitsu, Ricoh, Regent Peripherals.)

- DPI : (300) (400) (600) (---)

- Threshold Range:

Photocopy Degradation Record

For the simulated photocopy degraded document page, the set of parameters that control the degrading process for each document will be integrated into the database in the form of records. The record has the following fields :

Record Field Definitions:

- Document ID:
- Source Document ID:
  The source document page ID from which current document is generated.

- Degradation Model: (Baird) (ISL)

- Model Parameter list : The list of parameters values used in the model.
Fax Communication Channel Degradation Record

For the simulated fax degraded document page, the set of parameters that control the degrading process for each document will be integrated into the database in the form of records. The record has the following fields:

Record Field Definitions:

- Document ID:
- Source Document ID:
  The source document page ID from which current document is generated.
- Degradation Model: (Fano) (Pareto) (McCullough) (Fritchman)
- Model Parameter list : The list of parameters values used in the model.

Page Condition Record

This record includes attributes that describe the visual conditions (or qualities) of a given document page. The record has the following fields (attributes):

Record Field Definitions:

- Document ID:
  The document ID can also be thought of as a filename that uniquely identifies each document in the database.
- n-th photocopy: (noise-free) (1) (2) (3) (4) (5) (6) (7) ( ).
- n-the Fax transmission: (noise-free) (1) (2) (3) (4) (5) (6) (7) ( ).
- Visible salt/pepper noise of all sizes: (yes) (no).
- Visible Vertical streaks of all kinds: (yes) (no).
- Visible Horizontal streaks of all kinds: (yes) (no).
- Extra contents from other pages on the top edge: (yes) (no)
- Extra contents from other pages on the bottom edge: (yes) (no)
- Extra contents from other pages on the left edge: (yes) (no)
- Extra contents from other pages on the right edge: (yes) (no)
• Page contents warped (curved) on the left edge: (yes) (no).
• Page contents warped (curved) on the right edge: (yes) (no).
• Page contents blurred on the left edge: (yes) (no).
• Page contents blurred on the right edge: (yes) (no).
• Visible Rotation: (yes) (no).
• Page Rotation Angle (in radius):
  • Overall Quality: (Perfect) (Near perfect) (good) (average) (poor) (very poor).
    – Perfect == No for items 4-15.
    – Near perfect == No more then one yes for items 4 to 15.
    – Good == No more than one yes from 4-6; no more than one yes from 11-15.
    – Average == No more than one yes from 4-6; no more than two yeses from 11-15.
    – Poor == No more than two yeses from 4-6; no more than three yeses from 11-15.
    – Very Poor == Worse than poor.

Document General Information Record

This record includes a set of attributes that are common to all document pages within the database.

Record Field Definitions:

• Document ID:

• Document Language: (English)
The value for this field is English for this database. (This field is provided for upward compatibility with future databases that we or others might produce in languages other than English for e.g. Kanji, Arabic etc.)

• Document Type: (Technical) (Letter) (Memo) (News).
This field will take an value of all the document types included in the database. Initially, the database will include the following types of documents:
  – Scientific and technical journals/proceedings/reports
  – News magazines (if resource is available)
• Document Category:
The document files will either be scanned digitized noisy documents or synthesized
digitized noise-free documents or degraded. The field thus has the following values:

- Scanned grey-scale
- Scanned binary
- Synthesized (LaTeX, PostScript)
- Photocopy Degraded (synthesized)
- Fax Degraded (synthesized)

• Multiple Font Types: (yes) (no).
• Multiple Font styles: (yes) (no).
• Multiple Font Sizes: (yes) (no).
• Dominant Font Type: (Serif) (Sans-Serif).
• Dominant Character Spacing: (proportional) (fixed).
• Dominant Font Size (Pts): \(<\ 9\) (9-12) (13-18) (19-24) (25-36) (\(\geq\ 36\)).
• Dominant Font Style: (Plain) (Bold) (Italic) (Underline) (Other).

  Combination of font styles are allowed.

  The word 'dominant' is defined as the most frequently used font (type, style, size) in a
given object. An object can be a document page or a zone. If equally distributed
multiple font (types, styles, sizes) are presented, then text body font overrules the others.

Page Layout Record

This record describes the page orientation and column layout within the live matter area
of a given document page. The record has the following fields (attributes):

Record Field Definitions:

• Document ID: String

• Page Orientation: (Portrait) (Landscape).

  Portrait is the standard page orientation for all documents. Landscape is an alternate
  page orientation. Landscape may be used when the document page consists of wide
  tabular data or a multicolumn format that does not fit easily into the portrait mode.
• Max Number of Text Columns: Integer.
  The number of equal-width text column(s) within of the live matter area of the document page.

• Column Layout: (regular) (combined-columns).

• Page Header Present: (yes) (no).

• Page Footer Present: (yes) (no).

Publication Record

In general, the attributes for a technical journal page or new magazine page will be quite different from the those of a letter and memo. This attribute record is for journal document pages. This record has the following fields:

Record Field Definitions:

• Document ID:

• Publication Name:

• Volume Number: Integer.

• Issue Number: Integer.

• Publishing month/period: (Jan) (Feb) (Mar) (Apr) (May) (Jun) (Jul) (Aug) (Sep) (Oct) (Nov) (Dec) (Spring) (Summer) (Autumn) (Winter)

• Publishing year: Integer.

• Page Number: Integer.

• Multiple pages from the same article: (yes) (no)
  A flag indicating whether multiple document pages from the same article are included in the database. The document pages within the same article can be retrieved by reference to the journal/report name, volume number etc. of the page.

Journal/Report Page Content Record

In general, the attributes describe the content of a technical journal page will be quite different from the those of a letter and memo or new magazine page. We define the attribute record for journal document pages as follows. The record has the following fields (attributes):
Record Field Definitions:

- Document ID: A unique four character string
  The document ID will be used as a part of the file name that uniquely identifies each document in the database.

- Text Zone Present: (yes) (no).

- Non-ASCII symbol embedded within a text zone: (yes) (no).
  A flag indicating the existence of such a zone in the document page. Some text zone contains symbols that are not found on the standard keyboard.

- Table Zone Present: (yes) (no).

- Half-tone Zone Present: (yes) (no).

- Drawing Zone Present: (yes) (no).

- Math Zone Present: (yes) (no).

  We define a zone as a math zone, if the zone has one of the following qualities:

  1. A stand-alone mathematical formula within the page live matter area.

  2. A mathematic formula embedded within a text line such that not all symbols in the formula can be vertically separable. For example, $X^\alpha, X_\beta, \alpha / \beta$, $\lim_{x \to \infty} x = 0$ are vertically separable. Whereas, $X^2$, $\frac{1}{2}, \sqrt{x + y}$,

    $\lim_{x \to \infty} x = 0$

    are not vertically separable.

- Number of zones: Integer.

  A document page may be visually partitioned into a set of rectangularly shaped zones. This attribute gives the number of zones in a document page.

Bounding Annulus Record

We define an annulus as the area between two rectangular boxes (one enclosing the other.) An annulus will be represented as the two rectangular bounding boxes (the ‘inner’ and the ‘outer’) that encloses an area of a document page. The area can be the header area, the footer area, or the live matter area of a document page, or a zone of a document page. The inner bounding box will approximate the smallest rectangular box that contain the entire area content and nothing else. The outer bounding box will approximate the largest rectangular box that contain the entire area content and nothing else. All page headers, page footers, and page live matters are classified as zones. Thus, they have unique zone ids. An annulus record has the following fields:
Record Field Definitions:

- Document ID:
- Zone ID:
- Upper-left Row Coordinate (for inner bounding box): Integer.
- Upper-left Column Coordinate (for inner bounding box): Integer.
- Lower-right Row Coordinate (for outer bounding box): Integer.
- Lower-right Column Coordinate (for outer bounding box): Integer.

Journal Zone Attribute Record

This attribute record describes a set of attributes that are common to zones from a journal/report document page. The record has the following fields (attributes):

Record Field Definitions:

- Document ID: A unique four character string
  (The ID of the corresponding document page.)
- Zone ID:
  A unique two character string assign to the zone.
- Zone Area: (Header) (Live Matter) (Footer).
- Zone Content: The zone content can be classified into following types: (A compilation of the descriptions and examples of the zone labels are given in ISL document—The English Document Database Zone Label Definitions and Examples

  - Text
  - Text with Non-ASCII Character(s)
  - Mathematics formula
  - Table
  - Form
  - Half-tone zone
  - Ruling
  - Logo
  - Map
- Drawing
- Advertisement
- Announcement
- Handwritten
- other

- Zone Label: The zone label can have the following values:
  - None Text Zone
  - Abstract Body
  - Abstract Heading
  - Affiliation(s)
  - Article Submission Information
  - Article Title
  - Author(s)
  - Author Biography
  - Caption
  - Diploma Information
  - Drop Cap
  - Executive Abstract Heading
  - Executive Abstract Body
  - Footnote Heading
  - Footnote Body
  - Highlight
  - Keyword Heading
  - Keyword Body
  - List Item
  - Section Heading
  - Section Excerpt
  - Section Bulletin
  - Society Membership Information
  - Synopsis
  - Text Body
- Page header
- Page footer
- Page Number
- Pseudo-codes
- Reference Heading
- Reference list item

A further refinement to the page header and footer is being considered. The refinement includes:

- Page header:Author(s)
- Page header:Article Title
- Page header:Journal Name
- Page header:Publisher and Copy Right Information
- Page footer:Article Publication Information
- Page footer:Publisher and Copy Right Information
- Page footer:Article Submission Information

A text zone may also be labeled as ‘section Heading’, if it is a leading word or phrase that has a function as the header of the text body, and occupies one text line. For example, the heading may be an algorithm heading (e.g., ‘Algorithm II’), a theorem heading (e.g., ‘Theorem A’), a proof heading (e.g., ‘Proof:’), a remark heading (e.g., ‘Remark:’), etc..

- Font Information: (Similar to that of page font information.)

- Character Sequence Direction: (left-right) (right-left) (top-down) (bottom-up).
  This field gives the direction of characters within a text line when a page is oriented to up-right position. For example, for a landscaped oriented page, the page needed to be rotated to in up-right position.

- Character Orientation: (up-right) (rotated-right) (rotated-left).
  This field gives the orientation of characters within the text line(s) when the page is oriented to up-right position.

- Zone’s column number:
  This attribute describes the zone’s column location.
• Next Zone ID within the same thread: (---) (nil)

Each document page may be read in several different logical sequences (called a thread). In our current design, we use this attribute to indicate the reading sequence among the zones that constitutes a thread. "nil" is used to indicate the end of a thread.

• Ground Truth Document Format: The attribute has the following values:
  
  – Character sequences
  – Character with positions
  – Both formats

The following attributes may be added to the record.

• Excerpts from Ground Truth: For text zone, the first few words of the first line of the text will be given here. This is for the purpose of additional quality control of the corresponding ground truth reside in the database.

• Text line alignment:
  
  This field represents the text line(s) alignment style in the zone. The type of text line alignment can have the following values:
  
  – Left aligned
  – Center aligned
  – Right aligned
  – Justified
  – Indented
  – Indent/Hanging
  – List Formatted (for list items)
  – Floating

Letters/Memos Zone Record

This attribute record describes a set of attributes that are common to zones from a letter/memo document page. The record has the following fields (attributes):

Record Field Definitions:

• Document ID:
• Zone ID:

• Zone Area: (Header) (Live Matter) (Footer).

• Zone Content: The zone content can be classified into following types:
  
  – ASCII Text
  – Text with non-ASCII symbols
  – Logo zone
  – Other

• Zone Label: The zone label can have the following values:

  – Page Header
  – Fax Header
  – Subject Header
  – Page Number
  – Date
  – Salutation
  – Addresser Name
  – Addresser Address
  – Addresser Affiliation
  – Addresser Phone
  – Addressee Name
  – Addressee Address
  – Addressee Affiliation
  – Addressee Phone
  – Body Paragraph
  – Closing
  – Signature
  – CC
  – Footnote (p.s.)

• Font Information: (Similar to that of page font information.)
• Character Sequence Direction: (left-right) (right-left) (top-down) (bottom-up).
  This field gives the direction of characters within a text line when a page is oriented
to up-right position. For example, for a landscaped oriented page, the page needed to
be rotated to in up-right position.

• Character Orientation: (up-right) (rotated-right) (rotated-left).
  This field gives the orientation of characters within the text line(s) when the page is
oriented to up-right position.

• Zone's column number:
  This attribute describes the zone’s column location.

• Next Zone ID within the same thread: (……) (nil)
  Each document page may be read in several different logical sequences (called a thread).
  In our current design, we use this attribute to indicate the reading sequence among
the zones that constitutes a thread. “nil” is used to indicate the end of a thread.

• Ground Truth Document Format: The attribute has the following values:
  - Character sequences
  - Character with positions
  - Both formats

The following attribute may be added to the record.

• Excerpts from Ground Truth: For text zone, the first few words of the first line of the
text will be given here. This is for the purpose of additional quality control of the
 corresponding ground truth reside in the database.

• Text line alignment:
  This field represents the text line(s) alignment style in the zone. The type of text line
alignment can have the following values:
  - Left aligned
  - Center aligned
  - Right aligned
  - Justified
  - Indented
  - Indent/Hanging
  - List Formatted (for list items)
  - Floating