# A Clustering-Based Algorithm for Automatic Document Separation

Kevyn Collins-Thompson
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA USA
kct@cs.cmu.edu

Radoslav Nickolov
Microsoft Corporation
1 Microsoft Way
Redmond, WA USA
radonick@microsoft.com

## ABSTRACT
For text, audio, video, and still images, a number of projects have addressed the problem of estimating inter-object similarity and the related problem of finding transition, or 'segmentation' points in a stream of objects of the same media type. There has been relatively little work in this area for document images, which are typically text-intensive and contain a mixture of layout, text-based, and image features. Beyond simple partitioning, the problem of clustering related page images is also important, especially for information retrieval problems such as document image searching and browsing. Motivated by this, we describe a model for estimating inter-page similarity in ordered collections of document images, based on a combination of text and layout features. The features are used as input to a discriminative classifier, whose output is used in a constrained clustering criterion. We do a task-based evaluation of our method by applying it the problem of automatic document separation during batch scanning. This involves finding the transition points between documents in a continuous series of images. We evaluate separation performance using a probabilistic error metric, and obtain a page separation accuracy of 95.7% on our test collection.

## Keywords
Document Separation, Image Similarity, Image Classification, Optical Character Recognition

## 1. INTRODUCTION

The problem of accurately determining similarity between pages or documents arises in a number of settings when building systems for managing document image collections. For example, we are interested in clustering search results for queries on document image collections, or performing near-duplicate detection for indexing and other purposes.

Another such problem is automatically finding document separators in an ordered collection of images, such as ones coming from scanning or fax transmission. For example, we might wish to scan a large set of technical papers in a continuous "batch" but automatically detect and retain the original boundaries between possibly dozens of papers, saving each as a separate file for ease of browsing, searching and retrieval. Existing imaging applications require the user to either separate the papers manually and start a separate scanning job for each one; physically insert blank or special separator pages; or mark each separator while examining possibly hundreds of thumbnails looking for the first page of each paper. Since these options may be inconvenient or time-consuming, we seek an automated method to assist the user with reliable document separation.

In addition, a single document may not originate as one contiguous set of pages, but be scattered into several disconnected, ordered subsets that we would like to recombine. Such scenarios are not uncommon when scanning large volumes of paper: for example, one document may be accidentally inserted in the middle of another in the queue. In other cases, pages may have been accidentally omitted and we wish to alert the user to this case also. Such examples would normally be difficult to detect without tedious manual review.

Our overall approach is most closely related to video and text segmentation methods, which are summarized in Section 2. We use features based on layout, document structure, and topic concepts to discriminate between related and unrelated pages. This feature set is described in Section 3. These steps in turn are used to derive an overall page-set similarity measure for clustering disjoint groups of pages, as described in Section 4. We discuss our error metric and evaluate the quality of our separation algorithm in Section 5. Lastly, we discuss the strengths and weaknesses of our approach and possible areas for further study in Section 6.

## 2. RELATED WORK

To be clear on terminology: we adopt the term "document" to signify an ordered collection of images. A single image in such a document is termed a "page". Two pages are "related" if they come from the same underlying document and are "unrelated" otherwise. We use the term "separation" for the process of identifying the transitions between documents in an ordered sequence. Related work on text or video refers to the same problem as "segmentation" but this has a different meaning in the image recognition literature (e.g. "segmenting" a single page into separate layout or texture elements) and so we prefer "separation" instead.

The problem of finding topic boundaries or transition points in content has been studied in several projects for text, video, and spoken audio. For example, video segmentation, or 'shot break' detection, uses audio and image features to discriminate between different news stories. The Informedia project [Haupt95] is a notable example of such work. The TDT Story detection/segmentation [Allan98] track has also spawned work on identifying topic boundaries in text and spoken audio. For example, Beeferman et al [Beefer99] use an exponential model based on topicality and cue-word features to partition text into coherent segments. Earlier work of Hearst [Hearst94] on TextTiling used a cosine similarity measure as part of an algorithm to subdivide texts into multi-paragraph subtopics.

We are unaware of any published work on the related problem for document images: performing automatic document separation. Current document image management applications accomplish document separation by detecting manually placed separator (or "patch") pages. Separator pages may be blank, or coded with barcode or other special identifying markers. Separator pages take time to print, insert between each document, and remove afterwards. They also increase the volume of paper to be scanned, especially with many short documents. Some applications allow the operator to view a field of thumbnails in order the mark the initial pages by hand.

Also, unlike video, audio, or text segmentation, the pages in a document collection may also require re-ordering or recombining, either because the pages were presented out-of-order to the system originally, as with a batch scan, or because the pages are coming from a process such as full-text search. In either case, the problem is more difficult than just partitioning an ordered set: we must also do clustering. The clustering may be made easier because we can constrain based on the context, and we discuss this is Section 4.

Related work on document image similarity includes a document image classifier described by Shin and Doermann [Shin00]. This assigns a page image to one of 12 general classes of layout structure, such as cover page, form, and so on, but does not attempt to determine if two page images have come from the same document. Doermann et al [Doer97] describe a method for detecting duplicate and near-duplicate document images which does not rely on OCR, using a robust signature of shape codes based on representative lines of text. Doermann [Doer98] gives a survey of techniques for document image indexing and retrieval.

Document image understanding is the broad research area which covers methods for deriving structure from page images. Haralick [Hara94] gives a general survey of work on two major sub-areas: reconstructing geometric page layout and finding logical page structure. Our application makes indirect use of such methods to obtain page features, but otherwise document separation can be considered a higher-level problem.

There is a large body of existing work on estimating similarity among images in general, especially for color photographs. Huang and Rui [Huang97] give a comprehensive survey of general image retrieval techniques. In cases where a document contains very little text, or for sub-areas on a page, such image similarity methods may be applicable, but in this study we focus on text-intensive document images.

# 3. FEATURE SELECTION

The following general types of features form the feature space the model uses to discriminate between related and unrelated pages. These features are currently chosen based on our knowledge of the problem and did not involve a feature selection algorithm.

## 3.1 Document Structure

Pages may be related by means of being labeled with meta-information. The three types of meta-information features we use in this algorithm are headers, footers, and page numbers. We identify these features by cross-correlating text elements between pages, with each such feature being used as a component in the separation feature space.

For headers and footers, we define a region of interest at the page top and bottom. Within that region, we look for closely-matching lines of text between pages in that region. We form text element correlations spanning multiple pages, using these correlations as feature space elements. We detect page number correlations in a similar way.

These features require OCR processing. Because the text from OCR may contain errors, we use the approximate string matching technique described in [Collins01] when forming the text element correlations.

## 3.2 Layout Structure

For text intensive images, which is the main area of interest, we extract and use the text layout information. We aim to reconstruct the word and line structure of the original document. In its crudest form, which turns out to be rather effective, this process requires no more than the word bounding boxes. Our feature space is derived using

probability distributions of the following text elements:

1. Word height
2. Character width
3. Horizontal word spacing
4. Line spacing
5. Line indentation

Layout features can be obtained by image segmentation techniques, and do not require full OCR, although that is how we obtain them in our implementation.

## 3.3 Text Similarity

If reliable text from OCR is available, we can include a simple word-based "cohesion" measure between two pages. Similar to TextTiling [Hearst94], we use a vector space model where each page is represented by a vector of word frequencies, and the similarity measure is the normalized cosine between the word-vectors of the two pages. We exclude very common words, and stem words using Porter's algorithm. Because the text from OCR may contain errors, we only use the text similarity calculation if the OCR confidence is above a preset threshold. Otherwise, we set the text similarity to a value representing an 'indeterminate' state.

## 3.4 General Image Content Features

The model we describe does not include general image content features, except perhaps indirectly via OCR, but we mention them for completeness, since these tie in with a very large body of existing work on image similarity.

Some examples include using the color spectrum of color documents, and the lower-resolution components of a wavelet representation. An earlier version of our system used the latter and was most effective for documents with very little text, such as Powerpoint slides. A combination of image segmentation, applicable to both monochrome and color, and general image-based features, such as logo detection, could add significantly to accuracy if OCR is not available or the OCR results are poor. Adding such features is a direction of future work.

# 4. PAGE SIMILARITY AND DOCUMENT CLUSTERING

Our overall approach is to treat document separation as a constrained bottom-up clustering problem, using an inter-cluster similarity function based on the features defined in Section 3. We will define a similarity measure for each feature type and then show how these are combined to obtain the overall inter-cluster similarity measure. We then briefly describe the clustering algorithm itself.

## 4.1 Document Structure Similarity

Once we identify potential headers, footers, and page numbers, we want to distinguish true cross-page correlations from random matchings. Some more difficult cases in this area include:

- Common table headings being mistaken for page headers
- Headers or footers which alternate every other page, or missing page numbers, as in magazines
- Figure or section headings may be mistaken for page numbers
- Multiple page numberings such as date/timestamp combinations, which are especially common in faxes

To address this, we calculate a weight for each potential multi-page text element match using a standard correlation function between the sequence indices and the indices for the same pages in the original document. The correlation function has a large value if there is a strong linear relationship between the indices. The weight calculation includes a factor based on the average string length of the text in the correlation, reflecting the fact that matches between very short strings are less likely to be significant than those between longer strings.

Page number sequences are treated similarly, except that we look at the correlation between the page indices, and the page number values found. The weights give us a ranking for page number sequences. Given a page, we associate it with the top-ranked sequence in which the page is present.

The page number feature distance between two pages is defined as:

| | |
|---|---|
| 0 | if the pages belong to the same number sequence |
| 1 | if the pages belong to conflicting number sequences |
| 0.5 | otherwise |

The header/footer feature distance is defined as the cosine distance between the header/footer feature vectors. Each header/footer feature is weighted to reflect our confidence in its ability to discriminate pages.

## 4.2 Layout Structure Similarity

Given a page, we build a histogram for each of the five layout features listed in Section 3.2. Each histogram is then smoothed using a standard kernel function. We define the similarity measure for each layout feature as the distance between the page distributions for that feature. In our implementation we used the KL divergence as the inter-distribution metric.

## 4.3 Text Similarity

If enough reliable text from OCR exists for each page, we

use the normalized cosine between the page word vectors as the distance in the text content dimension.

## 4.4 Overall Page Similarity

The five layout structure similarity scores and the text similarity score are presented to a linear classifier, from which we estimate a posterior probability that the two pages are related. In our current implementation we use a Support Vector Machine (SVM). The SVM was trained on the six above features extracted from the training set described in Section 5.

The final similarity score between two pages is a simple decision rule using these steps:

1. If a reliable page number distance exists, this is returned as the page similarity score.
2. Else if a reliable header/footer similarity exists, this is returned as the page similarity score.
3. Else the page similarity score is the layout/text similarity estimate derived from the SVM.

In Section 5 we evaluate four different versions of this decision rule according to the presence or absence of the page numbering feature and header/footer feature.

## 4.5 Clustering

We perform bottom-up clustering, starting with each page in its own cluster, and then progressively merge pairs of clusters using a single-linkage criterion. Single-linkage defines the distance between clusters to be the distance between their most similar pages. The merging is constrained by only comparing pages within a specified threshold distance *d* in the overall page sequence. The distance between all other pages is defined to be infinity. We keep merging clusters, until the distance between the clusters in that topmost element exceeds a specified threshold.

# 5. EVALUATION

In this section we evaluate our page similarity measure according to the accuracy of the document separation task. We also look at the effectiveness of some sub-components of the overall similarity measure.

Our training set is a collection of 191 documents comprising 2709 document images. The test set is a collection of 70 documents comprising 980 document images. The test set documents were selected from a different image archive and taken 'blindly' without knowledge of their specific contents or formatting. Both the training and test sets use a wide variety of different layout styles, header and page number formats.

The test set appears to reflect a realistic variety of styles and is of reasonable difficulty. For example, approximately 40% of all documents do not use headers, footers, or page numbering. In addition, the image quality of about 20% of the pages is quite poor, resulting in no useful OCR text. About 50% of these "difficult" pages were also incorrectly auto-rotated during the OCR process, resulting in about 10% of the total page collection being upside-down, with several orientation changes being possible in a single document. Most of the documents from both sets are technical in nature and include machine learning and SIGIR papers, technical memos and diagrams, email, and product specifications. It seems realistic to assume that documents with related content are likely to occur during the batch scanning process. We have informally observed that people often archive entire folders or boxes at a time which were already classified according to specific subject areas.

Following the method used to measure text segmentation accuracy in Beeferman et al [Beefer99], we measure the effectiveness of the segmentation algorithm by calculating the probability that two pages selected from a batch of *N* document pages will each be placed into the correct document. In particular, we sweep across a page collection comparing pages a fixed distance of *k* pages apart, where *k* is one-half of the mean document size in pages. Each comparison looks at the number of separators between the two pages. If the ground-truth and test set have the same counts, we add a score of one to a counter, otherwise we give a score of zero and do not update the counter. After processing all test images we divide by the total number of comparisons to get the probability of correct separation.

We examined the effectiveness of each step in the final decision rule of section 4.4, i.e. the similarity measures for page numbers, header/footers, and layout. The results are shown in Table 1.

| Features Included | Separation Accuracy |
|---|---|
| Layout only | 89.25% |
| Layout + Page Number | **95.68%** |
| Layout + Header/Footer | 85.37% |
| Layout + Header/Footer + Page Number | 90.02% |

Table 1: Effect of various feature combinations on page separation accuracy

The best performance on our test set, a separation accuracy of 95.68%, was obtained by using page numbering similarity if available, or general layout/text similarity otherwise. Without page numbering, using only layout similarity, the accuracy dropped to 89.25%. To get a more precise interpretation of these differences, we would need to take into account the proportions of pages with each feature type; we omit this analysis for space reasons.

One surprising finding is that incorporating header/footer similarity, at least according to our decision rule, resulted in worse performance than not using it at all. In most cases, the errors were false separators. These were caused by short, two or three-page spurious correlations between

pages which had poorly recognized text, but otherwise were related and close in layout structure, such as the upside-down pages mentioned earlier. We believe header/footer detection is still important in discriminating between unrelated pages, for example, when separating different articles in a single magazine scan. Our results suggest that we need to reconsider the weighting and usage of header/footer features in our overall similarity measure, especially as a function of the page recognition confidence.

## 6. CONCLUSIONS

We have shown that reasonably accurate automatic document separation is possible using a combination of layout and text features. Our header/footer feature was more unreliable than expected, and our best performance on our test collection was achieved without it, using a decision rule which gives precedence to page numbering similarity, followed by layout structure similarity. Our results suggest that if OCR were not available, we could use layout features derived using image segmentation techniques, and still obtain good accuracy.

One improvement in future work would be to replace our current ad-hoc set of features with those derived from a well-defined selection step. Also, the SVM / decision rule combination could be replaced by a single trained discriminative classifier, such as a decision tree. In this way we could apply exponential models similar to those used for text. Finally, we would introduce general image content features to handle documents without recognized text. Based on our experience in this work, we hope to apply our similarity techniques to other document image-related problems such as clustering the results of a document image search.

## 7. ACKNOWLEDGMENTS

This material is partially based on work supported by NSF grant IIS-0096139. Any opinions, findings, conclusions, or recommendations expressed in this material are the authors', and do not necessarily reflect those of the sponsor. The authors would also like to acknowledge the work of Ming Liu and Daryl Lawton in creating an earlier prototype of automatic document separation. This was based on a different implementation but used some of the same layout and text features. We thank Jamie Callan for reviewing an earlier version of this paper.

## 8. REFERENCES

[Allan98] J. Allan, J. Carbonell, G. Doddington, J. Yamron, Y. Yang. Topic Detection and Tracking Pilot Study Final Report. *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[Beefer99] D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*: *Special Issue on Natural Language Processing*, 34 (1-3): 177-210, 1999. C. Cardie and R. Mooney (editors).

[Collins01] K. Collins-Thompson, C. Schweizer, S. Dumais. Improved string matching under noisy channel conditions. In *Proceedings of CIKM 2001*. Nov. 2001, pp 357-364.

[Doer97] D. Doermann, H. Li and O. Kia. The detection of duplicates in document image databases. In *Proceedings of the International Conference on Document Analysis and Recognition*, pp. 314-318, 1997.

[Doer98] D. Doermann. The indexing and retrieval of document images: A survey. Technical Report CS-TR-3876, University of Maryland, Computer Science Department, February 1998.

[Hara94] R.M. Haralick, Document image understanding: geometric and logical layout. *Proceedings of the 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* pp. 385-390. Seattle, WA 1994.

[Haupt95] A. G. Hauptmann, M. A. Smith. Text, Speech and Vision for Video Segmentation: The Informedia Project. *AAAI-95 Fall Symposium on Computational Models for Integrating Language and Vision.* November, 1995.

[Hearst94] M. A. Hearst. Multi-paragraph segmentation of expository text. *Proceedings of the 32nd Meeting of the Association for Computational Linguistics (ACL '94)*, June 1994. Las Cruces, NM, USA.

[Huang97] T. S. Huang and Y. Rui. Image retrieval: Past, present, and future. In *Proc. of Int. Symposium on Multimedia Information Processing*, Dec 1997.

[Shin00] C. Shin, and D. Doermann. Classification of document page images based on visual similarity of layout structures. In *Proceedings of SPIE Document Recognition and Retrieval VII 3967*, pp. 182-190, 2000.