STOCHASTIC ATTRIBUTED K-D TREE MODELING OF TECHNICAL PAPER TITLE PAGES

Song Mao

National Library of Medicine
Bethesda, Maryland 20894

Azriel Rosenfeld

Center for Automation Research
University of Maryland
College Park, Maryland 20742

Tapas Kanungo

IBM Almaden Research Center
San Jose, California 95120

ABSTRACT

Structural information about a document is essential for structured query processing, indexing, and retrieval. A document page can be partitioned into a hierarchy of homogeneous regions such as columns, paragraphs, etc.; these regions are called physical components, and define the physical layout of the page. In this paper we develop a class of models for the physical layouts of technical paper title pages. We model physical layout using hidden semi-Markov models for directional projections of page regions, and a stochastic attributed K-d tree grammar model for the $2D$ hierarchical structure of these regions. We use the models to generate sets of synthetic title page images of three distinctive styles, which we use in controlled experiments on page structure analysis.

1. INTRODUCTION AND PRIOR WORK

A document page can be partitioned into a hierarchy of physical components, such as pages, columns, paragraphs, textlines, words, tables, figures, halftones, etc. This structural information is essential for structured query processing, indexing, and retrieval the content of the document. Document understanding modules, such as Optical Character Recognition (OCR) and graphics recognition modules, can also be selectively applied to the structural components of document page images.

Title pages of technical papers contain rich bibliographical information about the papers, which is crucial for their indexing and retrieval. In this paper, we demonstrate how to analyze the layout of the physical components of technical paper title pages using hidden semi-Markov models [1] and a stochastic attributed K-d tree grammar.

Document structure analysis can be regarded as a syntactic analysis problem. The order and containment relations among the components of a document page can be described by an ordered tree structure and can be modeled by a tree grammar which describes the page at the component level in terms of regions or blocks. We will introduce a class of such grammars in Section 2.

A few researchers have developed document physical layout analysis algorithms that make use of grammatical methods. Kopec and Chou [2] describe an algorithm for segmenting a column of text that is modeled using a stochastic regular grammar, but their algorithm must be given templates for the symbols in the language and that the page is segmented into columns by some other procedure. Tokuyasu and Chou [3] used regular grammars to describe the structure of document page images in terms of axis-parallel rectangles, and used a Turbo decoding approach to estimate the 2D image from the observations, but they provided very limited experimental verification of their approach. Krishnamoorthy *et al.* [4] described an algorithm that constructs a tree in which each node represents an axis-parallel rectangle, but the segmentation and labeling process in their algorithm is based on heuristically specified parameters, not on estimated ones.

2. THE MODEL

Our physical layout model consists of two parts: (1) a hidden semi-Markov model that describes the grouping of lowest-level page regions (strips, in a given direction) into rectangular blocks; (2) a K-d tree grammar (defined below) that describe the hierarchical decomposition of the page into these blocks.

To parse a given page image, we first divide it into thin parallel strips and count the number of black pixels in each strip. The resulting sequence of pixel counts is taken to be the observation sequence of a hidden semi-Markov model. The state changes of the model then indicate boundaries between groups of strips. For example, a line of text contains a group of strips with high pixel counts, and the strips in a gap have low pixel counts. The states define labels of the groups of strips. These labels are vocabulary symbols of a stochastic attributed K-d tree grammar which we use to find possible physical layouts of the page. In the following description we assume $K = 2$.

The "productions" r_i of the grammar are directional subdivision processes, each of which is of the form

$$r_i : X_i \xrightarrow{\rho_i} \Psi_i.$$

Here Ψ_i is a set of trees defined on the vocabulary in which non-leaf nodes are labeled with nonterminal symbols and leaf nodes are labeled with either terminal or nonterminal symbols; ρ_i denotes the coordinate direction along which the subdivision takes place; and the children of each parent node in each tree in Ψ_i are ordered. The trees can be of two types: a terminating type, in which all the leaf nodes are labeled with terminal symbols, and a nonterminating type, in which one or more of the leaf nodes are labeled with nonterminal symbols.

Each symbol represents a rectangular region. The position and size of this region are defined by the pairs of coordinates of two of its opposite corners. The coordinates associated with the start symbol S represent the entire page, and the region associated with a parent node is the union of the regions associated with its children.

For each r_i, let $Left(r_i)$ and $Leaves(r_i)$ denote the left-side symbol X_i of r_i and the set of ordered sets of right-side leaf nodes of the trees in Ψ_i. r_i is applied in direction ρ_i to partition the rectangular region $D(Left(r_i))$ into a set of ordered sets of rectangular regions $D(Leaves(r_i))$. Associated with each node of each tree in Ψ_i is its coordinate (in that direction) relative to the coordinate of the region represented by $Left(r_i)$. Also associated with each leaf node of r_i are two features, black pixel count and size, which are used to determine which sets of strips can be grouped into the region associated with the leaf node.

A derivation in the grammar involves the application of a sequence of r_i's. In a generative derivation, the application of r_i attaches some tree in Ψ_i to a leaf node that has the label $Left(r_i)$. In a parsing derivation, the application of r_i joins a set of root nodes that have labels of $Leaves(r_i)$ to a new root node that has the label $Left(r_i)$. When r_i is applied and r_j is applied to the result (i.e., $Left(r_j) \in Leaves(r_i)$), the directions of r_i and r_j must be different.

The grammar is stochastic, as defined by the following probabilities:

- p_i, the probability of applying r_i; for any symbol A, $\sum_{Left(r_i)=A} p_i = 1$.
- For each r_i, each node of $Leaves(r_i)$ represents a group of strips. The process of grouping the strips into subregions is performed by a hidden semi-Markov model λ_i. The sequence of strip black pixel counts is taken to be the observation sequence of λ_i. The states of λ_i are vocabulary symbols of the grammar.
- $\lambda_i = (A_i, B_i, C_i, \pi_i)$, where
  - A_i denotes the state transition probability matrix of λ_i.
  - B_i is a matrix of the probabilities that the pixel count of a strip has a given value if the strip belongs to a given state.
  - C_i denotes the size (or duration) probability matrix: a matrix of the probabilities that the number of strips belonging to a given state has a given value.
  - π_i denotes the initial probability vector: the probability that λ_i starts in a given state.

An attributed *complete tree* is a tree whose root node is labeled S and has associated coordinates that represent the initial region, and whose leaf nodes are labeled with terminal symbols and have associated coordinates that define a partition of that region, as well as associated feature values. The *complete language* of the grammar is the set of attributed complete trees that can be created either in a generative derivation starting from a single node labeled S, or in a parsing derivation starting with a set of nodes that have terminal labels.

The probability of generating an attributed complete tree T in the grammar is a product of probabilities taken over the r_i's that are used in the deriving T. For each of these r_i's, we multiply p_i by the probability of the sequence of strips and feature values that are associated with the most probable set of leaf nodes of any tree in Ψ_i. Let $\mathbf{q}_i$ denote the partition of the strips of $D(Left(r_i))$ into groups of strips associated with one of the ordered sets in $Leaves(r_i)$ and let $\mathbf{o}_i$ be the vector of feature observations on the strips of $D(Left(r_i))$. Thus if a sequence $r_1, r_2, \ldots, r_k$, a sequence of feature value vectors $\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_k$, and a sequence of state vectors $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k$ are used to generate T, the probability of generating T is

$$\prod_{i=1}^{k} p_i P(\mathbf{o}_i, \mathbf{q}_i|\lambda_i) \quad = \quad \prod_{i=1}^{k} P(\lambda_i) P(\mathbf{o}_i, \mathbf{q}_i|\lambda_i)$$

where $S \overset{r_1,\mathbf{o}_1,\mathbf{q}_1}{\Longrightarrow} T_1 \overset{r_2,\mathbf{o}_2,\mathbf{q}_2}{\Longrightarrow} T_2 \ldots \overset{r_k,\mathbf{o}_k,\mathbf{q}_k}{\Longrightarrow} T$,

If we are given an image I, the feature observations ($\mathbf{o}$'s) on the strips of the leaf nodes in a derivation of I are fixed. However, there can be more than one derivation of I using different combinations of r's and $\mathbf{q}$'s. We can use the grammar to find a maximum-probability hierarchical partition (i.e., a maximum-probability parse) of I. To do this, we divide I into strips, and we consider all possible partitions of these strips into groups each of which corresponds to a leaf node of some tree in Ψ_i. With each r_i and each partition $\mathbf{q}_i$ we associate the probability $P(\lambda_i)P(\mathbf{o}_i, \mathbf{q}_i|\lambda_i)$. We find the sequence of r_i's and associated partitions for which the product is as great as possible (* denotes the optimum):

$$\begin{aligned} T^*(I) &= \arg \max_{r_1, r_2, \ldots, r_k, \mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k} \prod_{i=1}^{k} P(r_i) P(\mathbf{o}_i, \mathbf{q}_i|\lambda_i) \\ &= \arg \max_{\lambda_1, \lambda_2, \ldots, \lambda_k, \mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k} \prod_{i=1}^{k} P(\lambda_i) P(\mathbf{o}_i, \mathbf{q}_i|\lambda_i) \end{aligned}$$

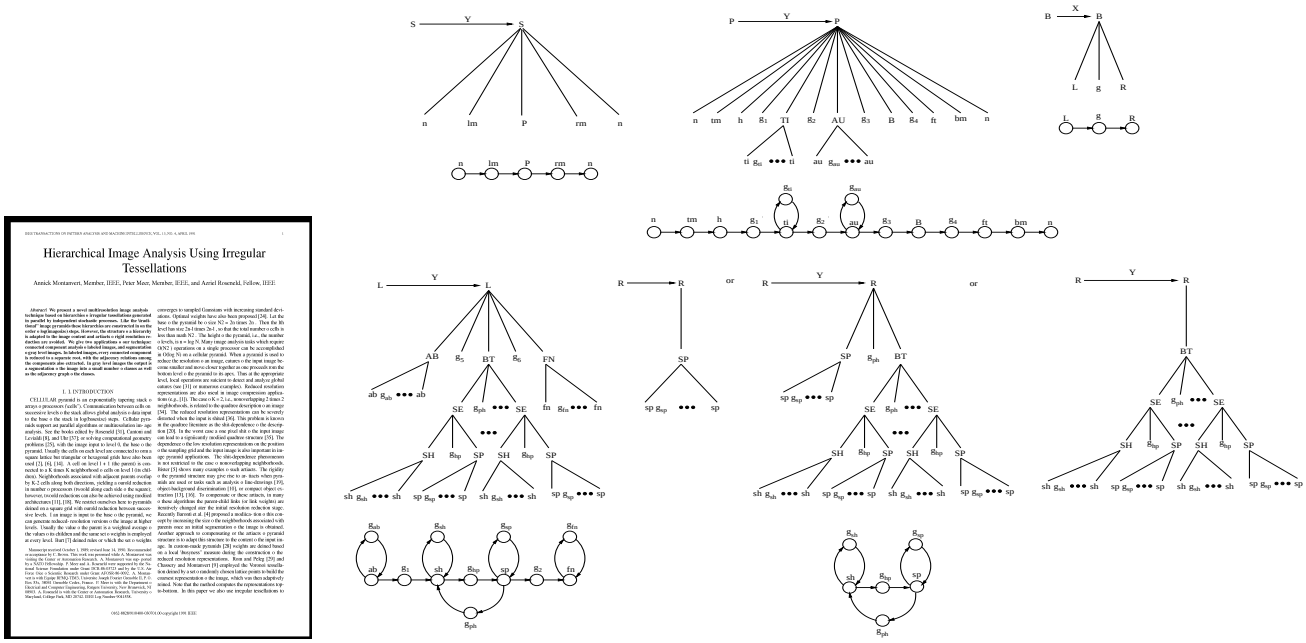


Fig. 1. A two-column title page image (a) and its physical layout model (b).

$$\approx \quad \arg \max_{\lambda_1,\lambda_2,\ldots,\lambda_k} \prod_{i=1}^{k} P(\lambda_i) P(\mathbf{o}_i, \mathbf{q}_i^* | \lambda_i)$$

The attributed complete tree T^* defined by this sequence, i.e., $S \stackrel{r_1,\mathbf{q}_1}{\Longrightarrow} T_1(I) \stackrel{r_2,\mathbf{q}_2}{\Longrightarrow} T_2(I) \ldots \stackrel{r_k,\mathbf{q}_k}{\Longrightarrow} T^*(I)$, specifies the maximum-probability hierarchical partition of the image. To find $T^*(I)$ we use a dynamic programming algorithm called the DV (for "duration Viterbi") algorithm. As we will see, this algorithm is much more powerful than the conventional Viterbi algorithm ("V algorithm"), in which state durations are not used (i.e. there is no C matrix).

Our model differs from non-grammar-based tree methods in the following aspects: 1) Our model is generative. 2) symbols are rewritten as sets of trees representing subdivisions of a region in a given direction. 3) the tree nodes in our grammar have associated coordinates which define K-dimensional rectangular regions. The coordinate aspect of our grammar makes it a very appropriate tool for generating and parsing Manhattan document layouts.

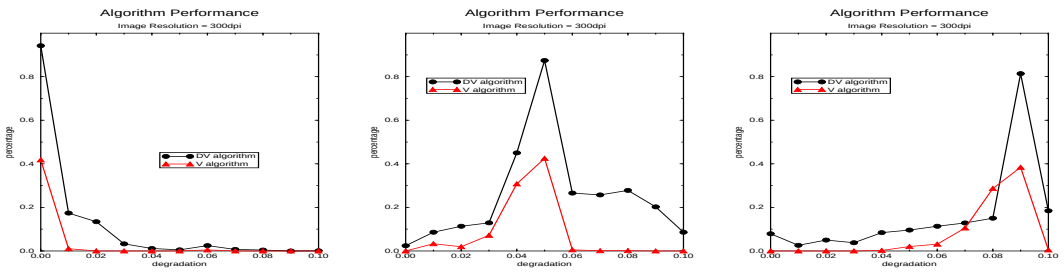
The physical layout analysis model for technical paper title pages of two-column format is shown in Figure 1. The descriptions of the symbols in the grammar are given in Table 1. Our performance metric was based on the fraction ρ of correctly detected textlines. Let l^H be a set of groundtruth textlines, each of which has a logical label. A textline is said to be correctly detected if it does not have any of the following six types of textline errors: 1) false dismissals: no segmented line significantly overlaps l^H; 2) false alarms: a segmented line does not significantly overlap any l^H; 3) merges: two or more groundtruth lines significantly overlap a segmented line; 4) cuts: l^H significantly overlaps both a segmented line and its complement; 5) excessive height: the segmented line that significantly overlaps l^H is too thick (vertically); and 6) incorrect labeling: the line is correctly segmented (on the basis of the significant overlap and height criteria), but is not labeled correctly. Since there are many textlines on a document page, measures based on textlines provide a statistically meaningful evaluation of performance.

Table 1. Symbol descriptions in the technical paper title page physical layout grammar. Note that the descriptions of the gaps are omitted.

Symbol Type	Description
Nonterminal	S: start symbol; P: main body of text; TI: title AU: author; B: two-column body; L, R: left, right column AB: abstract; BT: sections; FN: foot note SH: section heading; SP: section paragraph
Terminal	h: header; lm, rm, tm, bm : left, right, top, and bottom margin n: noise streak; ti: title line; au: author line; ft: footer line ab: abstract line; fn: footnote line sh: section heading line; sp: section paragraph line

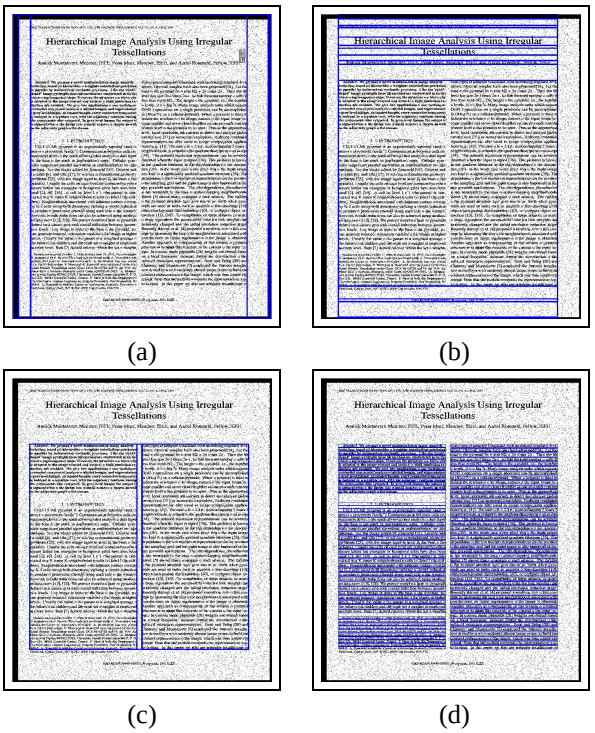
3. EXPERIMENTS

We conducted experiments on technical paper title pages that had three styles: one-column, mixed one- and two- column, and two-column. The one-column style is used by SPIE conferences; the mixed one- and two- column style is used by SICE conferences; and the two-column style is used by many IEEE transactions and conferences. We obtained LaTeXstyle files for these title page styles from the IEEE, SPIE, and SICE web sites.



(a) (b) (c)

Fig. 2. Performance using the V and DV algorithms and the 2-d tree grammar with model parameters estimated on noise-free training images (a) and on images degraded at two levels (b-c).



(a) (b) (c) (d)

Fig. 3. Segmentation of a noisy image into page (a), text body (b), columns (c), and textlines (d) using the DV algorithm and the 2-d tree grammar. The algorithm parameters were estimated on a training dataset with the same degradation level.

We used our stochastic generative document model to randomly generate a dataset of noise-free synthetic title page images with groundtruth for each of the three styles. The page text was taken from the symbolic text of the title pages in the University of Washington III dataset. Each page image was sampled at 300 dpi. Table 2 shows the dataset descriptions for the three styles.

We modified the DVI2TIFF software to generate clean images and their textline groundtruth, including the logical label of each line. Each page in the training datasets was degraded at two degradation levels using the document degradation model of [5]. As can be seen from Figure 3, even the lower of the two degradation levels is quite substantial.

Table 2. Dataset descriptions for three layout styles.

Style	Training Dataset	Test Dataset
one-column	13 pages	16 pages
mixed one- and two-column	11 pages	10 pages
two-column	9 pages	8 pages

Each page in the test datasets was degraded at ten degradation levels. Two of these levels were the same as the levels used for training. Figure 3 shows an example of the segmentation of a two-column title page into page, text body, columns, and textlines using the K-d tree grammar and the DV algorithm. Figure 2 shows evaluation results using the V and DV algorithms on a set of title page images that had all three layout styles, using a combined grammar.

The performance using the DV algorithm is significantly better than that for the V algorithm in nearly all cases. In all cases, both algorithms attain the best performance at the noise level used for algorithm training.

4. REFERENCES

[1] D. R. Cox and H. D. Miller, *The Theory of Stochastic Processes*, Methuen and Co LTd, London, 1965.

[2] G. E. Kopec and P. A. Chou, "Document image decoding using Markov source models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, pp. 602–617, 1994.

[3] T. A. Tokuyasu and P. A. Chou, "Turbo recognition: A statistical approach to layout analysis," in *Proceedings of SPIE Conference on Document Recognition and Retrieval*, San Jose, CA, January 2001.

[4] M. Krishnamoorthy, G. Nagy, S. Seth, and M. Viswanathan, "Syntactic segmentation and labeling of digitized pages from technical journals," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 737–747, 1993.

[5] T. Kanungo, R. M. Haralick, and I. Phillips, "Nonlinear local and global document degradation models," *International Journal of Imaging Systems and Technology*, vol. 5, pp. 220–230, 1994.