Learning to Segment Document Images

K.S. Sesh Kumar, Anoop Namboodiri, and C.V. Jawahar

Centre for Visual Information Technology,
International Institute of Information Technology, Hyderabad, India

Abstract. A hierarchical framework for document segmentation is proposed as an optimization problem. The model incorporates the dependencies between various levels of the hierarchy unlike traditional document segmentation algorithms. This framework is applied to learn the parameters of the document segmentation algorithm using optimization methods like gradient descent and Q-learning. The novelty of our approach lies in learning the segmentation parameters in the absence of groundtruth.

1 Introduction

Document image layout has a hierarchical (tree like) representation, with each level encapsulating some unique information that is not present in other levels. The representation contains the complete page in the root node, and the text blocks, images and background form the next layer of the hierarchy. The text blocks can have a further detailed representations like text lines, words and components, which form the remaining layers of the representation hierarchy. A number of segmentation algorithms [1] have been proposed to fill the hierarchy in a top-down or bottom-up fashion. However, traditional approaches assume independence between different levels of the hierarchy, which is incorrect in many cases. The problem of document segmentation is that of dividing a document image ($\mathcal{I}$) into a hierarchy of meaningful regions like paragraphs, text lines, words, image regions, etc. These regions are associated with a homogeneity property $\phi(\cdot)$ and the segmentation algorithms are parameterized by θ.

The nature of the parameters θ depend on the algorithm that is used to segment the image. Most of the current page segmentation algorithms decide the parameters θ, apriori, and use it irrespective of the page characteristics as shown in [2]. There are also algorithms that estimate the document properties (like average lengths of connected components) and assign values to θ using a predefined method [3]. Though these methods are reasonably good for homogeneous document image collections, they tend to fail in the case of wide variations in the image and its layout characteristics. The problem becomes even more challenging in the case of Indian language documents due to the presence of large variations in fonts, styles, etc. This warrants a method that can learn from examples and solve the document image segmentation for a wide variety of layouts. Such algorithms can be very effective for large and diverse document image collections such as digital libraries. In this paper, we propose a page segmentation algorithm based on a learning framework, where the parameter vector θ is learnt from a set of example documents, using homogeneity properties of regions, $\phi(\cdot)$. The feedback from different stages of the segmentation process is used to learn the parameters that maximize $\phi(\cdot)$.

S.K. Pal et al. (Eds.): PReMI 2005, LNCS 3776, pp. 471–476, 2005.

Hierarchical framework: The hierarchical segmentation framework is characterized at each level i, by a parameter set $\theta_i, 1 \leq i \leq n$. The input to the segmentation algorithm is the document image, ($\mathcal{I}$), which is generated by a random process, parameterized by θ. Given an input document $\mathcal{I}$, the goal is to find the set values for parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$, that maximizes the joint probability of the parameters for a given ($\mathcal{I}$), using (1).

$$\hat{\theta} = arg \max_{\theta_1,\theta_2,\ldots,\theta_n} P(\theta_1, \theta_2, \ldots, \theta_n|\mathcal{I}) \tag{1}$$

$$= arg \max_{\theta_1,\theta_2,\ldots,\theta_n} P(\theta_1|\mathcal{I}).P(\theta_2|\mathcal{I}) \ldots P(\theta_n|\mathcal{I}) \tag{2}$$

$$= arg \max_{\theta_1,\theta_2,\ldots,\theta_n} P(\theta_1|\mathcal{I}).P(\theta_2|\mathcal{I}, \theta_1) \ldots P(\theta_n|\mathcal{I}, \theta_1, \ldots, \theta_{n-1}) \tag{3}$$

On assuming independence between different levels of the hierarchy, the parameters are conditionally independent of each other as they characterize the segmentation process at each level. Hence (1) can be rewritten as (2). However, this is not a valid assumption as errors at a level of the hierarchy is propagated downwards deteriorating the overall performance of the segmentation system. Hence the segmentation algorithm and its parameters at the lower levels depend on the parameters of the upper levels and the input image $\mathcal{I}$. To incorporate this dependency into the formulation, we need to rewrite (1) as (3).

To achieve optimal segmentation the joint probability of the parameters $P(\theta_1, \cdot\cdot\theta_n)$ needs to be modeled. However, the distribution is very complex in case of page segmentation algorithms. In addition a large set of algorithms that can be used at different levels of hierarchy. Hence a fitness measure of segmentation is defined that approximates the posterior probability at each stage. Our goal is to compute the parameter θ_i that maximizes the posterior $P(\theta_i/\mathcal{I}, \theta_1, \theta_2, \ldots, \theta_{i-1})$ or the fitness function at level i.

Conventionally, segmentation has been viewed as a deterministic partitioning scheme characterized by the parameter θ. The challenge has been in finding an optimal set of values for θ, to segment the input Image $\mathcal{I}$ into appropriate regions. In our formulation, the parameter vector θ is learned based on the feedback calculated in the form a homogeneity measure $\phi(\cdot)$ (a model based metric to represent the distance from ideal segmentation). The feedback is propagated upwards in the hierarchy to improve the performance of each level above, estimating the new values of the parameters to improve the overall performance of the system. Hence the values of θ are learned based on the feedback from present and lower levels of the system. In order to improve the performance of an algorithm over time using multiple examples or by processing an image multiple times, the algorithm is given appropriate feedback in the form of homogeneity measure of the segmented regions. Feedback mechanisms for learning the parameters could be employed at various levels.

2 Hypothesis Space for Page Segmentation

A document image layout is conventionally understood as a spatial distribution of text and graphics components. However, for a collection of document images, layout is characterized by a probability distribution of many independent random variables. For example, consider a conference proceedings formatted according to a specific layout style

sheet. The set of values for word-spaces, line spaces etc. will follow a specific statistical distribution. For a new test page, segmentation implies the maximization of the likelihood of the observations (word-spaces etc.) by assuming the distribution. In our case, the objective is to maximize the likelihood of the set of observations by learning the parameters of the distribution, indirectly in θ-space.

We assume the existence and possibly the nature of a distribution of the region properties, but not the knowledge of the ground truth. For the problem of page segmentation, assumption of an available distribution (ground-truth) may not be very appropriate. Hence the problem is formulated as learning of the parameter vector θ, which minimizes an objective function $J(\mathcal{I}, \phi, \theta)$. The function $J(\cdot)$ could be thought of as an objective quality measure of the segmentation result. The parameter θ includes all the parameters in the hierarchy of document segmentation, $\theta_1, \cdots, \theta_n$. In this case, the objective function, $J(\cdot)$, can be expanded as a linear combination of n individual functions. Objective functions can be defined for different segmentation algorithms based on the homogeneous properties of the corresponding regions it segments.

Segmentation Quality Metric: A generic objective function should be able to encapsulate all the properties associated with the homogeneity function ϕ. In the experiments, a function is considered that includes additional properties of the layout, such as density of text pixels in the segmented foreground and background regions and a measure that accounts for partial projections in multi-column documents. In this work, the inter line variance (σ_1), the variance of the line height (σ_2), the variance of the distance between words (σ_3), the density of foreground pixels within a line (ILD) and between two lines (BLD), the density of foreground pixels within a word(IWD) and between words (BWD) are considered. We use the following objective functions $J(\mathcal{I}, \phi_l, \theta_l)$ and $J(\mathcal{I}, \phi_w, \theta_w)$, that needs to be maximized for best line and word segmentation respectively:

$$J(\mathcal{I}, \phi_l, \theta_l) = \frac{1}{1+\sigma_1} + \frac{1}{1+\sigma_2} - BLD + ILD \tag{4}$$

$$J(\mathcal{I}, \phi_w, \theta_w) = \frac{1}{1+\sigma_3} - BWD + IWD \tag{5}$$

The value of each of the factors in the combination in the above equations falls in the range $[0, 1]$, and hence $J(\mathcal{I}, \phi_l, \theta_l) \in [-1, 3]$ and $J(\mathcal{I}, \phi_w, \theta_w) \in [-1, 1]$.

3 Learning Segmentation Parameters

The process of learning tries to find the set of parameters, θ, that maximizes the objective function $J(\mathcal{I}, \phi, \theta)$: $\arg\min_{\theta} \; J(\mathcal{I}, \phi, \theta)$. Let $\tilde{\theta}_i$ be the current estimate of the parameter vector for a given document. We compute a revised estimate, $\tilde{\theta}_{i+1}$, using an update function, $D_\theta(.)$, which uses the quality metric $J(\cdot)$ in the neighborhood of $\tilde{\theta}_i$ to compute $\delta\theta_i$.

$$\tilde{\theta}_{i+1} = \tilde{\theta}_i + \delta\theta_i; \quad \delta\theta_i = D_\theta(J(\mathcal{I}, \phi, \tilde{\theta}_i)) \tag{6}$$

In order to define the parameter vector θ, we need to look at the details of the segmentation algorithm that is employed. The algorithm used for segmentation operates in

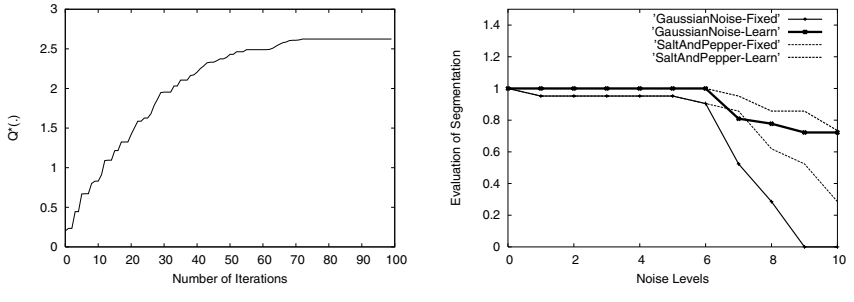
two stages. In the first stage, image regions are identified based on the size of connected components (size $\geq \theta_c$) and are removed. The remaining components are labeled as text. The second stage identifies the lines of text in the document based on recursive XY cuts as suggested in [1]. The projection profiles in the horizontal and vertical directions of each text block are considered alternately. A text block is divided at points in the projection, where the number of foreground pixels fall below a particular threshold, θ_n. In order to avoid noise pixels forming a text line, text line with a height less than a threshold, θ_l is removed. The lines segmented are further segmented into words using a parameter θ_w. We also restrict the number of projections to a maximum of 3, which is sufficient for most real-life documents. One could employ a variety of techniques to learn the optimal set of values for these parameters ($\theta_c, \theta_n, \theta_l$ and θ_w).

The parameter space corresponding to the above θ is large and a simple gradient-based search would often get stuck at local maxima, due to the complexity of the function being optimized. To overcome these difficulties, a reinforcement learning based approach called Q-learning is used. Peng et al. [4] suggested a reinforcement learning based parameter estimation algorithm for document image segmentation. However, our formulation is fundamentally different from that described in [4] in two ways: 1) It assumes no knowledge about the ground truth of segmentation or recognition, and 2) The state space is modeled to incorporate the knowledge of the results of the intermediate stages of processing unlike in [4] that uses a single state after each stage of processing.

3.1 Feedback Based Parameter Learning

The process of segmentation of a page, is carried out in multiple stages such as separation of text and image regions, and the segmentation of text into columns, blocks text lines and words. The sequential nature of the process lends well to learning in the Q-learning framework.

In Q-learning, we consider the process that is to be learned as a sequence of actions that takes the system from a starting state (input image $\mathcal{I}$) to a goal state (segmented image) [5]. An action a_t from a state s_t takes us to a new state s_{t+1} and could result in a reward, r_t. The final state is always associated with a reward depending on the overall performance of the system. The problem is to find a sequence of actions that



(a) (b)

Fig. 1. (a): $Q(\cdot)$ over iterations, (b): segmentation accuracy of our approach and that of using fixed parameters in the presence of noise

maximizes the overall reward obtained in going from the start state to the goal state. The optimization is carried out with the help of the table ($Q(s_t, a_t)$) that maintains an estimate of the expected reward for the action a_t from the state, s_t. If there are n steps in the process, the Q table is updated using the following equation:

$$Q(s_t, a_t) = r_t + \gamma r_{t+1} + \ldots + \gamma^{n-1} r_{t+n-1} + \gamma^n \max_a(Q(s_{t+n}, a)) \quad (7)$$

In the example of using XY cuts, the process contains two stages, and the above equation would reduce to $Q(s_t, a_t) = r_t + \gamma \max_a Q(s_{t+1}, a)$, where r_t is the immediate reward of the first step. Figure 1(a) shows the improvement of $Q^*(s_t, a_t) = \max_a Q(s_t, a_t)$ value over iterations for a particular document image. We note that the Q function converges in less than 100 iterations in our experiments.

4 Experimental Results

The segmentation algorithm was tested on a variety of documents, including new and old books in English and Indian languages, handwritten document images, as well as scans of historic palm leaf manuscripts collected by the Digital Library of India (DLI) project. Figure 2 shows examples of segmentation on a palm leaf, handwritten document, an indian language document with skew and a document with complex layout. Note that the algorithm is able to segment the document with multiple columns and images even when there is considerable variation between the font sizes, inter-line spacing among the different text blocks and unaligned text lines.

Performance in presence of Noise: Varying amounts of Gaussian noise and Salt-and-Pepper noise were added to the image and segmentation was carried out using the parameters learned from the resulting image. Figure 1(b) shows the segmentation performance of the learning-based approach as well as that using a fixed set of parameters, for varying amounts of noise in the data. We notice that the accuracy of the learning

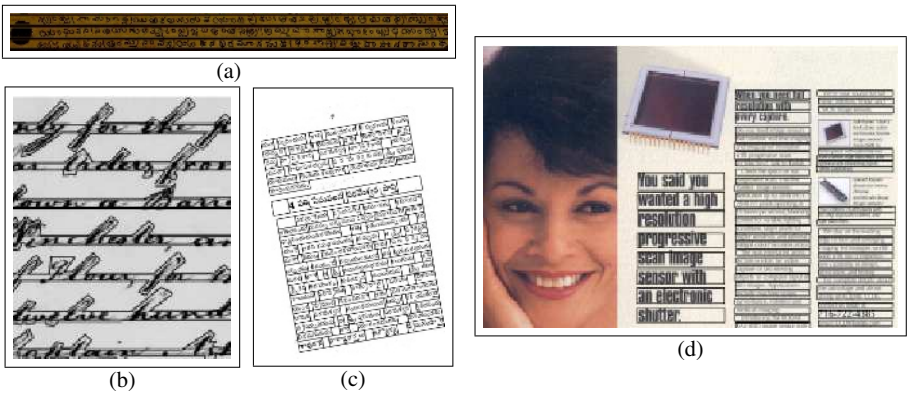


Fig. 2. Results of segmentation on (a) palm leaf, (b) handwritten document, (c) line and word segmentation of Indian language document with skew and (d) a multicolumn document

approach is consistently higher than that using a fixed set of parameters. At high levels of noise, the accuracy of the algorithm is still above 70%, while that using a fixed set of parameters falls to 20% or lower. The accuracy of segmentation, ρ, is computed as $\rho = (n(L) - n(C_L \cup S_L \cup M_L \cup F_L))/n(L)$, where $n(.)$ gives the cardinality of the set, L is the set of ground truth text lines in the document, C_L is the set of ground truth text lines that are missing, S_L is the set of ground truth text lines whose bounding boxes are split, M_L is the set of ground truth text lines that are horizontally merged, and F_L is the set of noise zones that are falsely detected [2]. We notice that the algorithm performs close to optimum even in presence of significant amounts of noise. Meanwhile, the performance of the same algorithm, when using a fixed set of parameters, degrades considerably in presence of noise.Our algorithm gave a performance evaluation of 97.80% on CEDAR dataset with five layouts and five documents of each layout and 91.20% on the Indian language DLI pages.

Integrating Skew Correction: Skew correction is integrated into the learning framework by introducing a skew correction stage to the sequence of processes in segmentation, after the removal of image regions. An additional parameter, θ_s, would be required to denote the skew angle of the document, which is to be rectified. The results of segmentation with skew can be viewed in 2(c). Defining an immediate reward for the skew correction stage, computed from the autocorrelation function of the projection histogram helps in speeding up the learning process.

The proposed algorithm is able to achieve robust segmentation results in the presence of various noises and a wide variety of layouts. The parameters are automatically adapted to segment the individual documents/text blocks, along with correction of document skew. The algorithm can easily be extended to include additional processing stages, such as thresholding, word and character segmentation, etc.

5 Conclusion

We have proposed a learning-based page segmentation algorithm using an evaluation metric which is used as feedback for segmentation. The proposed metric evaluates the quality of segmentation without the need of ground truth at various levels of segmentation. Experiments show that the algorithm can effectively adapt to a variety of document styles, and be robust in presence of noise and skew.

References

1. G.Nagy, S.Seth, M.Vishwanathan: A Prototype Document Image Analysis System for Technical Journals. Computer **25** (1992) 10–12
2. Mao, S., Kanungo, T.: Emperical performance evaluation methodology and its application to page segmentation algorithms. IEEE Transactions on PAMI **23** (2001) 242–256
3. Sylwester, D., Seth, S.: Adaptive segmentation of document images. In: Proceedings of the Sixth International Conference on Document Analysis and Recognition, Seattle, WA (2001) 827–831
4. J.Peng, B.Bhanu: Delayed reinforcement learning for adaptive image segmentation and feature extraction. IEEE Transactions on Systems, Man and Cybernetics **28** (1998) 482–488
5. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. The MIT Press (1998)