A hierarchical analysis scheme for robust segmentation of Document Images using white-spaces

Ritu Garg, Gaurav Harit and Santanu Chaudhury
Indian Institute of Technology, Delhi. Hauz Khas, New Delhi, India
Email: {ritu2721a, gharit, schaudhury}@gmail.com

***Abstract*—Page segmentation is an important step in document image processing. Major applications need to provide intelligent access to documents contaning both text components and non-text components like pictures, graphics, tables etc. This paper presents an approach for document image segmentation using white space existing around text/image blocks in conjunction with morphological masking operations. The proposed framework analyzes the white space using a series of morphological opening operations. This produces a hierarchy of blocks formed using white space aggregation. The proposed segmentation technique is applicable to the class of documents with Manhattan Layout. Experimental results have confirmed that the proposed framework produces desirable results without the need of any threshold for pixel aggregation, unlike many algorithms of this class.**

I. Introduction

There is a widespread and increasing interest in converting paper based document information in electronic format. Document image analysis and interpretation deals with a variety of documents such as letters, books, newspapers, magazines etc available in a variety of scripts/languages and formats. Hence, document image understanding, particularly transformation of paper documents to their electronic version has become an important and a challenging application domain for researchers.

Page segmentation is done to analyze the layout and content of the document page. Various page segmentation techniques have been proposed which are typically applied to binary images. Traditional methods in document image interpretation proceed from *bottom-up* i.e. computation of connected components and merging them in to regions on the basis of local evidences and then deriving information from these data-structure.

Several classical approaches [1], [2], [3], [4] exist and all these methods are threshold sensitive. A well known approach for page segmentation is connected component aggregation [5]. Mandal et. al. use a hierarchical structure using local and global features of documents for segmentation. Mighlani et. al. [6] use color information to hierarchically segment document images into regions of interest represented as polygons. Though hierarchical analysis has been previously used for connected component aggregation of text information, we believe that the approach presented in this paper is novel in the sense that it deals with hierarchical aggregation of *white spaces* for doing segmentation.

The primary limitation of many existing techniques [7], [8], [9] [10] is that they are more or less heuristical, specific to a particular application, and assume a predefined layout of the document page. The approach proposed in this paper is an enhancement of our earlier work on Bottom-Up segmentation [11]. The proposed methodology obviates the dependence of the algorithm on pixel aggregation thresholds. Development of a methodology which can handle a variety of formats of document images with Manhattan Layout, without adjusting any threshold is the motivation of this work.

The organization of this paper is as follows. Section 2 describes the the earlier bottom-up approach which we have improved upon in this paper. We particularly highlight the failure points and give examples and discussions on why it has failed. Section 3 onwards we discuss the proposed scheme followed by experimental results in section 4. Concluding remarks are made in section 5.

II. Bottom-Up Segmentation

In this section we describe the bottom-up segmentation as reported in [11]. The algorithm works by aggregating the white spaces surrounding the component blocks. It then marks out rectangular blocks using white spaces as separators.

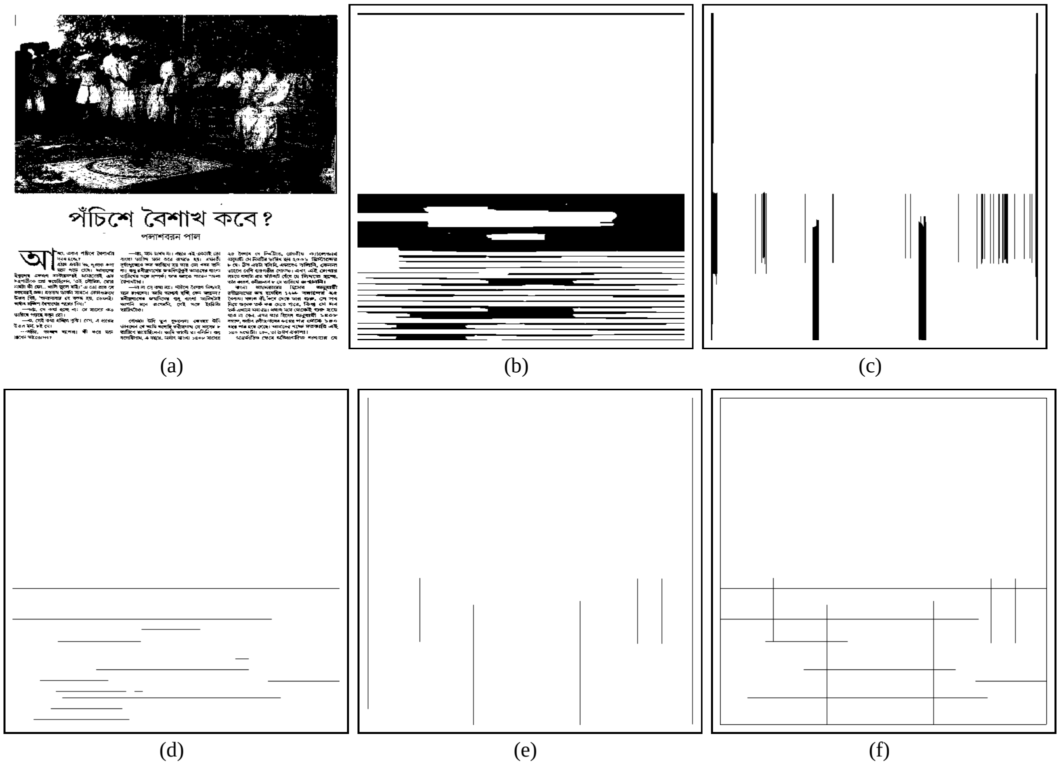
The algorithm can be briefly outlined in following steps:

(a) A given document image in gray-scale is first binarized. White spaces usually surround the document components (see figure 1(a)) in binarized images and thus qualify as reliable measures of separating one element block from another.

(b) Runs of white spaces greater than $1/5^{th}$ of the dimensions of the page are identified both in horizontal and vertical directions. Figure 1(b) and (c) shows white spaces marked as black lines.

(c) Next these thick runs are thinned using a thinning algorithm. The thinned lines are shown in figure 1(d) and (e). If there are broken lines or over-lapping lines quite close to one another, they are merged to form a single line. This finally gives a mesh of horizontal and vertical lines, as shown in figure 1 (f).

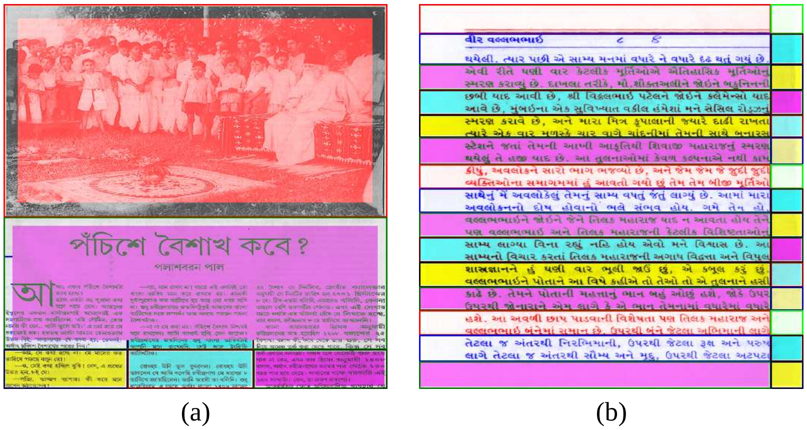
(d) We identify rectangular blocks out of this mesh. The algorithm used for identifying rectangular blocks is listed below.



(a) (b) (c)

(d) (e) (f)

Fig. 1. Illustration of Bottom-Up Segmentation Algorithm for a Bengali Document Image



(a) (b)

Fig. 2. Figure (a) Shows example of Over-Segmented Document Image (b) Shows example of Under-Segmented Document Image

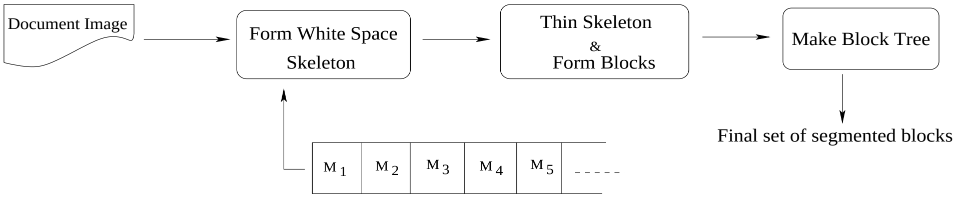

Document Image
Form White Space Skeleton
Thin Skeleton & Form Blocks
Make Block Tree
Final set of segmented blocks
M 1
M 2
M 3
M 4
M 5

Fig. 3. Flow Diagram for the Proposed Scheme for Segmentation

1. Mark all the crossings (intersections) of horizontal and vertical lines.
2. Pick up the closest possible pair of vertical crossings on the given horizontal line satisfying an eligibility criterion that atleast one crossings in the pair has not been used to form any other block. If none of the crossings are found to be eligible, pick up the next horizontal line and follow step 2, else go to step 3.

3. Try to traverse a smallest possible rectangle in the anti- clockwise direction with the two vertical crossings as edges. If all possibilities of traversing a rectangle fail, then go go to step 2. If a rectangle is formed note the block dimensions, mark those vertical crossings as used and go to step 2.
4. If there is any left over region which has not been covered by any block, then form a new block to cover it.

The above algorithm is sensitive to thresholds and the supplied values for certain parameters. For example the minimum run length of white spaces to be identified for a separator was specified as 100, the threshold of allowable pixel distance for merging horizontal or vertical lines was specified as 20. The choice of a threshold can give satisfactory results for a class of documents but it generally leads to over-segmentation or under-segmentation for other types of documents. Figure 2(a) shows an example of over-segmentation where a single block has been divided into multiple blocks. Improper merging of horizontal lines has resulted in under-segmentation as seen in figure 2(b). Thus it is difficult to generalize this algorithm for different classes of documents because of sensitivity to thresholds.

III. Proposed Methodology for Hierarchical Segmentation

To overcome the original algorithm's dependence on threshold while giving satisfactory segmentation results, we describe our proposed methodology in this section. The flow of the algorithm is summarized in figure 3. The steps of the algorithm are listed as follows:-

1 Let $\mathcal{M}_1$, $\mathcal{M}_2$, ..., $\mathcal{M}_z$ be a set of morphological masks. The masks are chosen in the increasing order of size. The algorithm does a morphological erosion [12] on the white pixels in the document image. We refer to the set of all white pixels as the *white-skeleton* of the document. It is to be noted that it is *not* the thinned structure resulting from a skeletonization operation. We perform the erosion step-by-step using the sequence of masks $\mathcal{M}_1$ to $\mathcal{M}_z$. For applying a mask, say $\mathcal{M}_i$, we do as follows:

(a) $\mathcal{W}_i \longleftarrow \begin{cases} \text{white pixels in binarized image, if } i = 0 \\ \mathcal{S}_i \quad \text{otherwise} \end{cases}$

(b) Perform Morphological opening operation on the set of pixels $\mathcal{W}_i$ with mask $\mathcal{M}_i$. The set of pixels covered by the mask as a result of this operation is denoted as $\mathcal{S}_i$. Hence, $\mathcal{S}_i$ denotes a skeleton of white pixels when using mask $\mathcal{M}_i$.

After applying all the masks $\mathcal{M}_1$, $\mathcal{M}_2$, ..., $\mathcal{M}_z$ we have a set of white skeletons $\mathcal{S}_1$, $\mathcal{S}_2$, ..., $\mathcal{S}_z$. The morphological erosion in a way progressively refines (removes *thin* details) these skeletons to restrict further analysis to only the important details. We consider that a thicker region of the white space is more likely to be a separator between bigger meaningful blocks.

2 In this step the white skeletons $\mathcal{S}_1$, $\mathcal{S}_2$, ..., $\mathcal{S}_z$ are thinned. We seek thinned lines which are straight, particularly we seek the horizontal and vertical lines. Next we identify rectangular blocks using these horizontal and vertical lines. This is done using the rectangle seeking algorithm given in [11].

Let the set of rectangular blocks identified using white skeleton $\mathcal{S}_k$ be $\mathcal{B}_k$.

$$\mathcal{M}_1, \longrightarrow \mathcal{S}_1 \longrightarrow \mathcal{B}_1$$
$$\mathcal{M}_2 \longrightarrow \mathcal{S}_2 \longrightarrow \mathcal{B}_2$$
$$\mathcal{M}_z \longrightarrow \mathcal{S}_z \longrightarrow \mathcal{B}_z$$

Hence we get sets of blocks $\mathcal{B}_1$, $\mathcal{B}_2$, ..., $\mathcal{B}_z$ for different masks used. Essentially $|\mathcal{B}_1| \geq |\mathcal{B}_2| \geq |\mathcal{B}_3| \geq \ldots$ where $|\mathcal{B}|$ denotes the cardinality of the set.

Given the individual blocks

$$\{b_1^0, b_1^1, b_1^2, b_1^3, \ldots\} \in \mathcal{B}_1$$
$$\{b_2^0, b_2^1, b_2^2, b_2^3, \ldots\} \in \mathcal{B}_2$$

the blocks in $\mathcal{B}_1$ would be contained in atleast one block in $\mathcal{B}_2$. In other words the set $\mathcal{B}_1$ consists a finer segmentation of the blocks in $\mathcal{B}_2$

3 Using hierarchical analysis we construct a tree for the segmented blocks. A block b_k^i (i^{th} block for k^{th} skeleton $\mathcal{S}_k$) would further get decomposed into child blocks for a skeleton $\mathcal{S}_j$ for some $j < k$.

We arrange the blocks in the form of a tree. The above algorithm can be explained with help of figure 5. Let $\mathcal{M}_1$, $\mathcal{M}_2$, ..., $\mathcal{M}_5$ represent the mask of different sizes such that,

$$\mathcal{M}_1 < \mathcal{M}_2 <, \ldots, < \mathcal{M}_5.$$

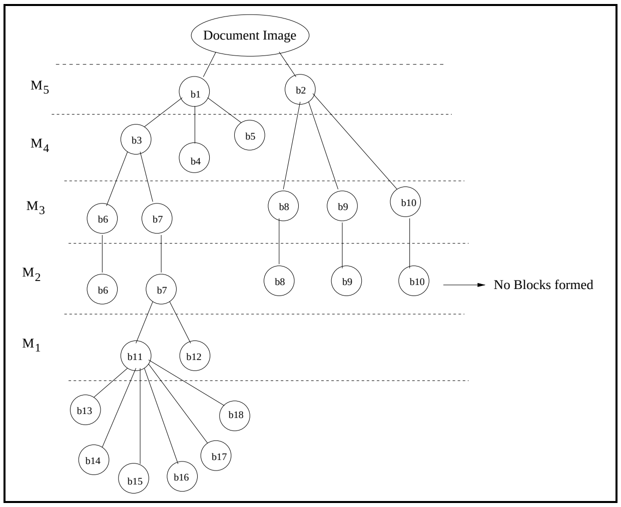
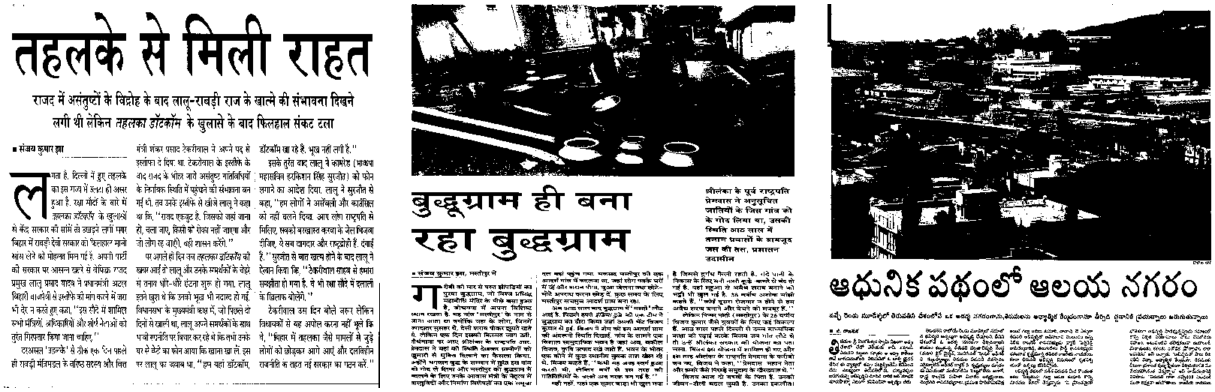


Fig. 5. Example block tree structure

Analysis of the document image is carried out starting with the largest sized mask being used. Assuming $z = 5$, $\mathcal{M}_5$ is the largest mask. An erosion with $\mathcal{M}_5$ would give the white skeleton $\mathcal{S}_5$. Let the rectangular blocks identified using this skeleton be b_1 and b_2, which are shown as the child nodes of the document image(root node) in figure 5. Application of smaller masks gives finer segmentations. For example, mask $\mathcal{M}_4$ yields blocks $\{b_3, b_4, b_5\}$ as sub-blocks within b_1 and mask $\mathcal{M}_3$ yields $\{b_6, b_7\}$ within b_3 and $\{b_8, b_9, b_{10}\}$ within b_2. Contin-



तहलके से मिली राहत

राजद में असंतुष्टों के विद्रोह के बाद लालू-राबड़ी राज के खात्मे की संभावना दिखने लगी थी लेकिन तहलका डॉटकॉम के खुलासे के बाद फिलहाल संकट टला

बुद्धग्राम ही बना रहा बुद्धग्राम

ఆధునిక పథంలో ఆలయ నగరం

(a) Document Image 1 (b) Document Image 2 (c) Document Image 3

Fig. 4. Shows example Document Images

uing this way, the smallest mask $\mathcal{M}_1$ yields blocks $\{b_{13}, b_{14}, \ldots, b_{18}\}$ with b_{11}.

Segmentation of a text block can happen at the article level, paragraph level, line level and word level. As we analyze with a smaller mask size the segmentation is likely to take place at the paragraph, line or word level. Thus the block hierarchy tree gives the organization of segmented rectangular blocks in a page.

Morphological operations applied to an image constitute a significant computational cost. The complexity is of the order $O(nm)$ where n is the number of pixels in the image and m is the size of the mask in pixels. We have implemented a speed up technique for a faster implementation of morphological erosion whereby we pre-compute and store the set of pixels where a mask can visit around a given pixel. This avoids the need of examining the mask on every white pixel in the image. This reduces the complexity by a multiplicative factors which actually varies (upto m) with image content. Further, since we perform erosion with a series of masks starting from the smallest size, a larger sized mask is applied only to those pixels which have been covered by the smaller sized mask.

IV. Experimental Results

In this section we present the results of segmentation with the proposed hierarchical analysis scheme. Figure 4 shows three example document images with – only text content, and text/image content. The results of our segmentation (i.e. the hierarchical block-tree) is shown in Figs 6, 7 and 8 respectively. In Figs 6, 7 and 8 the third row shows the identified blocks before applying the morphological operator. Individual blocks have been shown in a distinct color. The fourth row shows the blocks after decomposition. Child blocks are shown in the same color as the parent. The hierarchical segmentation produced by our scheme has successfully segmented the document image into a kind of logical decomposition. The largest mask identifies the image blocks, text blocks (articles, article headings). A smaller sized mask identifies the individual paragraphs, text columns and headline words. Further decreasing the mask size gives the line/word level segmentation. Our proposed scheme is able to handle documents with a variety of font sizes. The hierarchical decomposition finally achieves a word segmentation for text blocks of any font size. We have applied our segmentation algorithm to our dataset of scanned pages in several Indian languages. Within each language category we had document pages of different genres – magazines, manuscripts and newspapers. Our segmentation output was quite satisfactory in terms of the correctness of the hierarchical decomposition. The only exceptions were some degraded images where binarization into correct foreground and background could not be obtained.

V. Conclusion

In this paper we have proposed a page segmentation scheme using progressive applications of morphological masking operations. Our scheme does not use thresholds and gives satisfactory results over a variety of document images. The document page layout gets represented in a tree structure giving decomposition at article, paragraph, and line or word level. For our experiments we have used square-shaped masks of different sizes. It would be an interesting study to see if anisotropic masks could be more suitable compared to isotropic masks.

Acknowledgment

This work was funded by the grant RP01894 from Ministry of Communication and Information Technology, Govt. Of India.

References

[1] Haralick, R.M., Shapiro, L.G.: Image Segmentation Techniques. In: Computer Vision, Graphics and Image Processing. Volume 29. (1985) 100 – 132

[2] Kohler, R.: A segmentation system based on thresholding,. In: Computer Vision, Graphics and Image Processing. Volume 15. (1981) 319 – 338

[3] Pal, N.R., Pal, S.: A review on image segmentation techniques. Pattern Recognition **26** (1993) 1277 – 1294

[4] Otsu, N.: A threshold selection method from gray level histograms. IEEE Transactions on Systems Man and Cybernetics **9** (1979) 62 – 66

[5] Antonacopoulos, A.: Page Segmentation Using the Description of the Background. Computer Vision and Image Understanding **70** (1998) 350 – 369

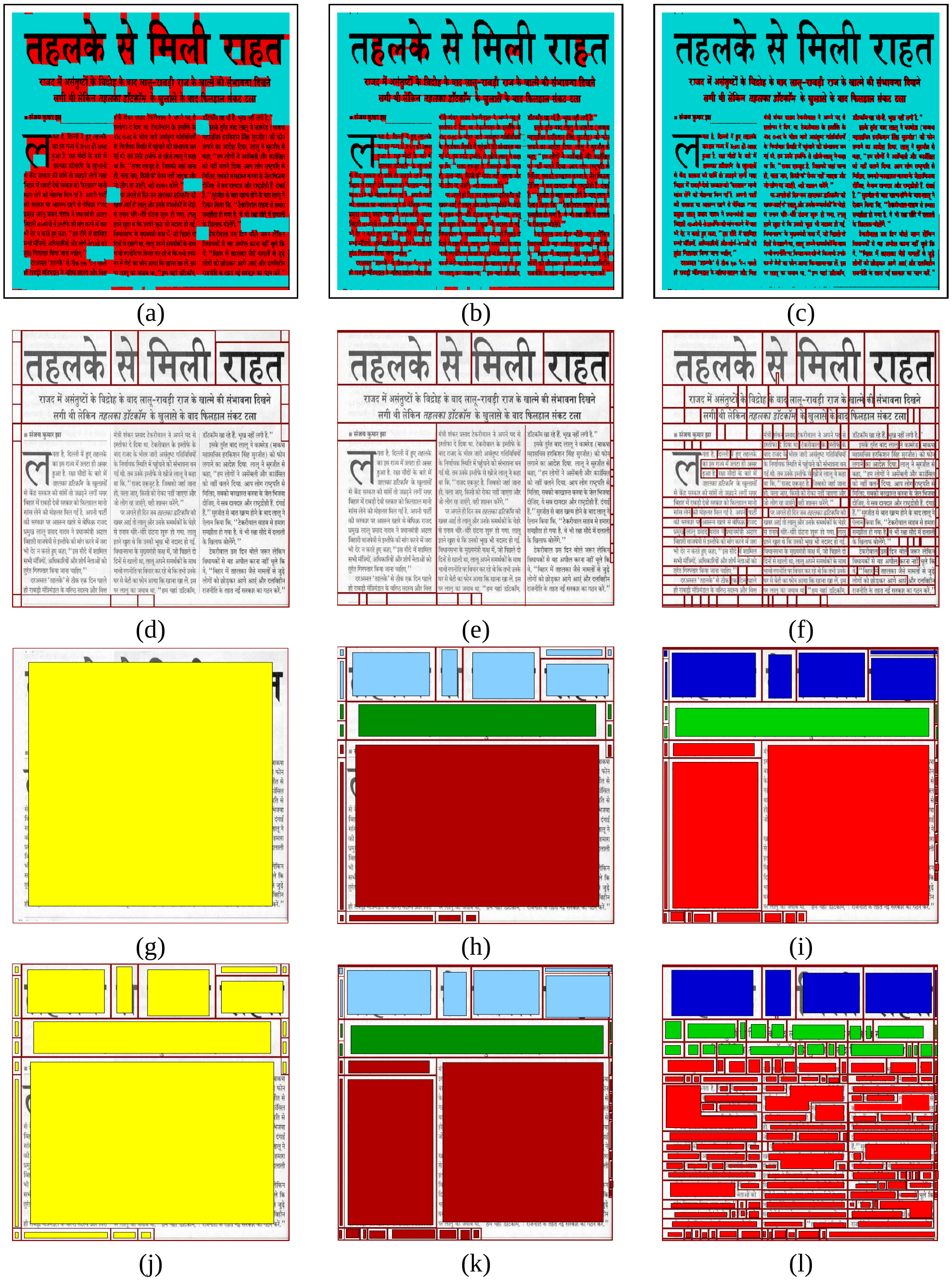

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l)

Fig. 6. Results of the Modified Bottom-Up Segmentation. Parts (a), (b), (c) show the regions covered by the mask of size 16x16 , 8x8 and 4x4 respectively; Parts (d), (e), (f) show the identified rectangular segmented blocks ; Parts (g), (h), (i), (j), (k), (l) show the segmented block hierarchy. The program has colored the child block with the same color as that of the parent block . The figure is best viewed in color. Please refer the soft copy of the paper.

[6] Mighlani, D.; Hennig, A.S.N.W.R.: Intelligent hierarchical layout segmentation of document images on the basis of colour content. In: 10th Annual Conference. Speech and Image Technologies for Computing and Telecommunications. Volume 1. (1997) 191 – 194

[7] Mandal, S., Chowdhuri, S., Das, A., Chanda, B.: Automated Detection and Segmentation of Form Document. In: Proceedings of the 5th International Conference on Advances Pattern Recognition (ICAPR2003), Calcutta, India (2003) 284 – 288

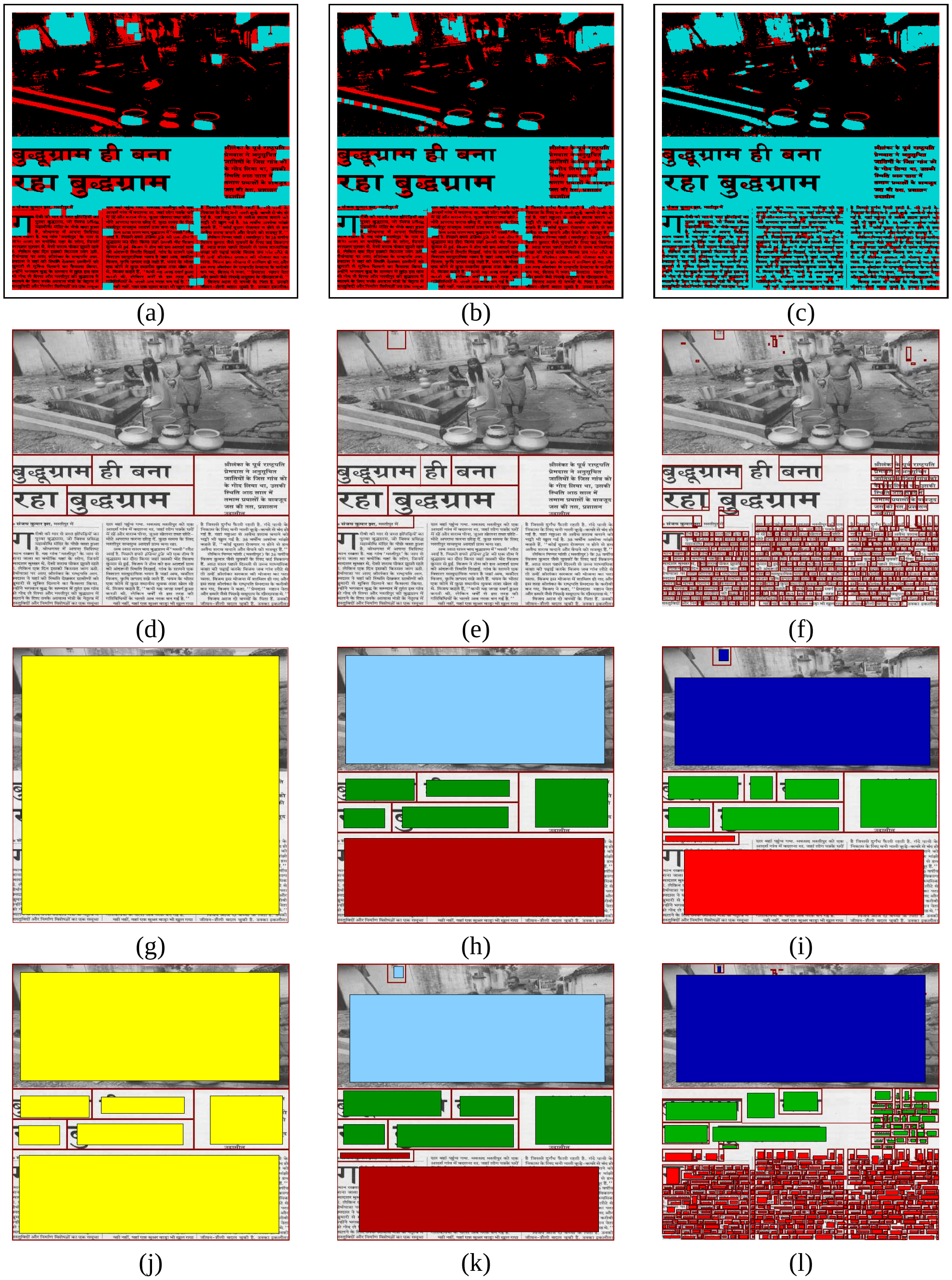
[8] Das, A., Chanda, B.: Segmentation of Text and Graphics in Document Image: A Morphological Approach. In: Proceedings of the International Conference on Computational Linguistics, Speech and Document Processing (ICCLSDP'98), Calcutta, India (1998) A50 – A56

[9] Das, A., Chowdhuri, S., Chanda, B.: A Complete System for Document Image Segmentation. In: Proceedings of national Workshop on Computer Vision, Graphics and Image Processing (WVGIP2002), Madurai, India (2002) 9 – 16

[10] Cheriet, M.; Said, J.S.C.: A recursive thresholding technique for image segmentation. IEEE Trans. on Image Processing, **7** (1998) 918 – 921

[11] Harit, G., Chaudhury, S., Gupta, P., Vohra, N., Joshi, S.: A Model Guided Document Image Analysis Scheme. In: Proc. ICDAR. (2001)

[12] Gonzalez, R.C., Woods, R.E.: Digital Image Processing. second edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1992)



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Fig. 7. Results of the Modified Bottom-Up Segmentation. Parts (a), (b), (c) show the regions covered by the mask of size 16x16 , 8x8 and 4x4 respectively; Parts (d), (e), (f) show the identified rectangular segmented blocks ; Parts (g), (h), (i), (j), (k), (l) show the segmented block hierarchy. The program has colored the child block with the same color as that of the parent block . The figure is best viewed in color. Please review the soft copy of the paper.

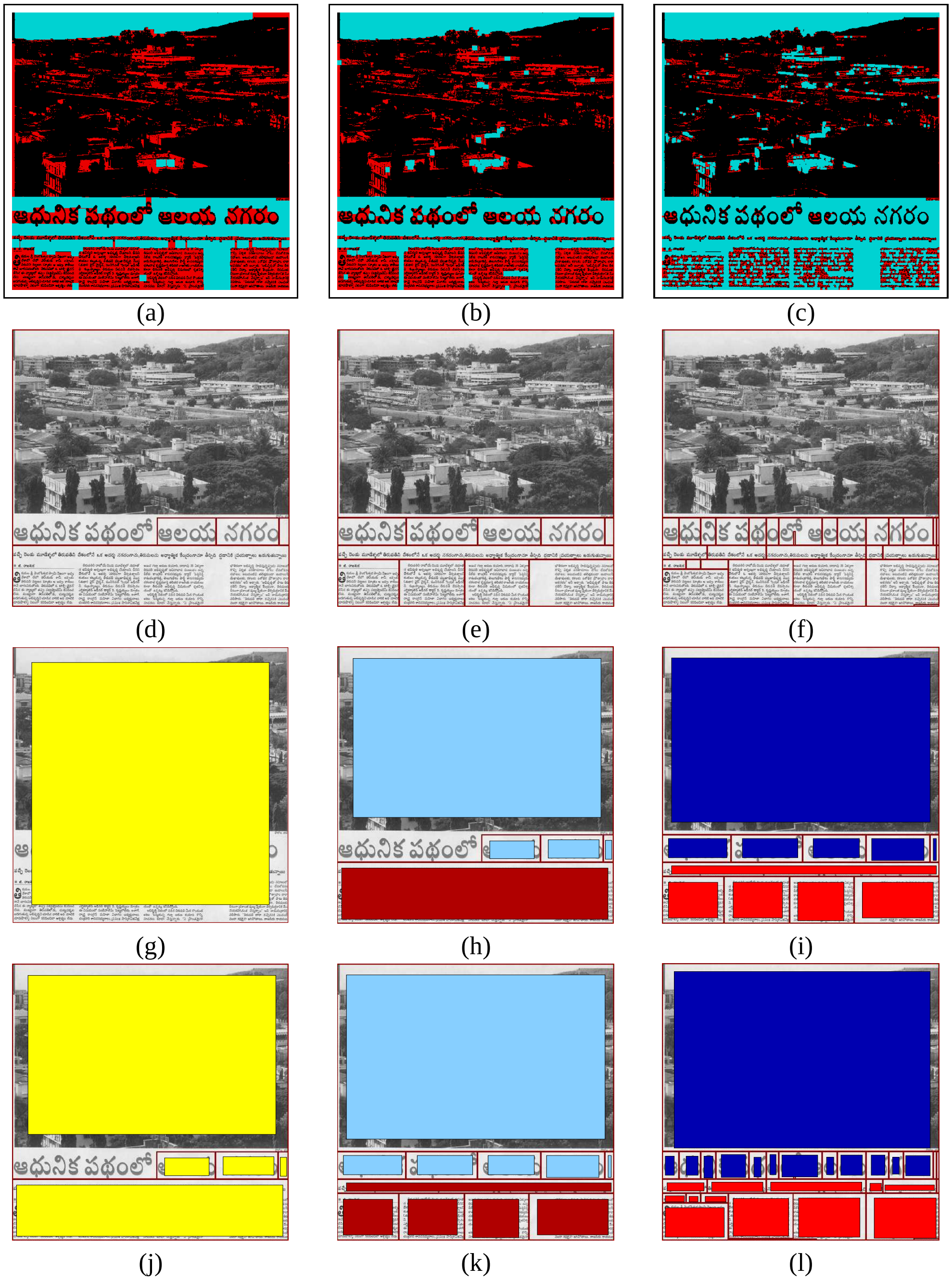

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l)

Fig. 8. Results of the Modified Bottom-Up Segmentation. Parts (a), (b), (c) show the regions covered by the mask of size 16x16 , 8x8 and 4x4 respectively; Parts (d), (e), (f) show the identified rectangular segmented blocks ; Parts (g), (h), (i), (j), (k), (l) show the segmented block hierarchy. The program has colored the child block with the same color as that of the parent block . The figure is best viewed in color. Please review the soft copy of the paper.