Transcoding of Document Images for Mobile Devices

Tabassum Yasmin
Electrical Engineering Department
IIT Delhi, New Delhi, India - 110016
tabuiitd@yahoo.com

Santanu Chaudhury
Electrical Engineering Department
IIT Delhi, New Delhi, India - 110016
santanuc@ee.iitd.ac.in

Richa Jain
Electrical Engineering Department
IIT Delhi, New Delhi, India - 110016
richaajain@yahoo.com

Abstract

In this paper we have presented a scheme for transcoding document images for presentation on handheld devices like PDA's, e-books etc. We have proposed technqiues suitable, in particular ,for images of documents of Indian languages having Devanagari based scripts (viz. Hindi, Marathi, Bengali, Assamese, etc). Appropriate compression scheme for textual component of document images exploiting script specific characteristics has been suggested. We have also explored use of the knowledge of the document model represented through standard ontology language for generation of document summary. An experimented system has been developed for validation of these schemes.

1. Introduction

Scanned legacy documents are being made web accessible.However,volume of the data that needs to be delivered for document images(even after compression) is very high. This demands reasonably large bandwidth for effective,smooth and jitter free interactive access. Further,availability of sufficient memory at the client end is another requisite for interactive browsing of large document images. Limitations of small screen size for hand-held devices also hampers readability of these documents. In this paper,we have proposed a set of document image transcoding techniques for overcoming these problems. Application of these transcoding techniques will facilitate web enabled access for document images using low cost PDA like computing devices having low bandwidth(may be modem based) internet connection.

Transcoding techniques transform content of a particular media type into a suitable representation for meeting the characteristics of the presentation device. Transcoding can be done apriori at the server or at a proxy [12]. Different technqiues have been suggested for image transcoding [9],text transcoding [11], video transcoding,[1] and audio transcoding [13]. Semantic transcoding [12] deals with the semantic content of data and use content analysis tools, annotations and meta-knowledge for transforming the content. However, the problem of document image transcoding has not been investigated much in the past. In [10] transcoding techniques involving word reflow for presenting document images on PDA's have been presented. The problem of document summarization from image content has been addressed in [3]. In this paper, we present transcoding techniques suitable, in particular, for images of documents of Indian languages having Devnagari based scripts (viz. Hindi, Marathi, Bengali, Assamese, etc.). A scheme has been formulated for word reflow using line and word segmentation scheme exploiting script specific characteristics. These script based characteristics have been also used for developing a word model based compression and transcoding scheme for the textual component of the document images. A scheme has been suggested for presentation of the image component of the document images. We have explored use of the knowledge of document model represented through standard ontology language for generation of the document summary.

2. Transcoding Scheme

The transcoding scheme makes use of an efficient model guided document image segmentation scheme [5]. The scheme involves in the first stage bottom-up segementation using image based features. Subsequently, regions obtained through bottom-up segmentation are classified into text and image regions using wavelet based features. Finally, using a model of the document page the result of bottom up segmentation is refined and logical components of the page are identified. We use segmented and labelled components of document pages as input to the transcoding scheme. Image and textual component of the document pages are transcoded differently. Knowledge about the document structure and labelled components are used for summary generation. User accesses different components based upon the summary and does not need to download the whole document image. Individual components are delivered after

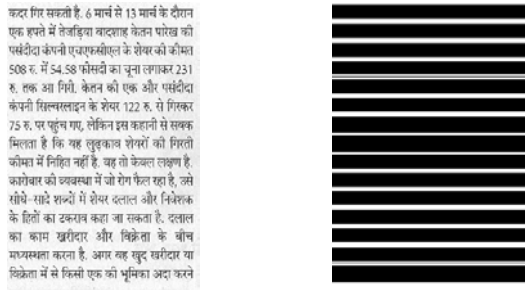
being appropriately transcoded.

3. Transcoding of Text Component

In this section, we present the transcoding scheme for the text part of the document image. Here we have assumed that there does not exist suitable OCR for automatic conversion of the image into electronic form. Our transcoding scheme treats the text part as an image and accordingly transforms the content. The transcoding scheme consists of two parts. First, the text part is segmented into words using image and script specific characteristics. Using these word images text is reformatted to suit the display size of the presentation device. This is called word reflow. We have also developed a novel compression scheme for textual component using graphical model of the words.

3.1. Word Re-flow

The word reflow scheme involves further processing of the textual components obtained by our segementation scheme. The segments are further broken down into lines. Each line is sub-divided into words. This process for documents in Devnagari script is different from that of English beacuse of the presence of top horizontal bar (shirorekha) and top and bottom ligatures. We use horizontal projection profile for line segmentation assuming that the page is skew corrected. We use the scheme proposed in [2]. Typically projection profile has a minima at the spacing between the lines. Between two minima a sharp peak is observed which corresponds to the top bar of the line.Using a suitable threshold minima is detected for separating out lines.But this results in segmenting the top/bottom ligatures as distinct lines.The decision rule is modified by incorporating a check on the height of the line. Results of line segmentation is shown in Fig 1. Words are obtained from the line using gaps in vertical projections of the line image.The results of word segmentation are shown in fig 2.



(a) Text block image (b) Line Segmentation

Figure 1. Example of Line Segmentation

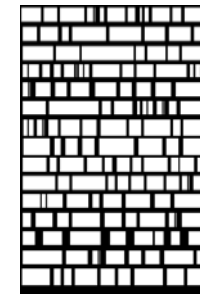


Figure 2. Word Segmentation

Word segmentation transforms the textual component of a raw document image into a form that is reflowable depending on the display size. Text is assumed to be left aligned. For each line, word images are floated one after another with a fixed space between them. Whenever addition of a new word causes the line width to exceed screen size the word is reflowed to the next line automatically. Size of the word images also can be changed by image transformation algorithms. The advantage of the word reflow scheme is that the textual component can be displayed on any presentation device preserving the original font. This is possibly for the first time a word reflow scheme for Devnagari script based document images have been implemented. Results of word reflow scheme are shown in Fig 3. Here screen width is 250 pixel. Fig 4 shows the same text region reflown for screen sizes of 220 pixels and 180 pixels respectively.

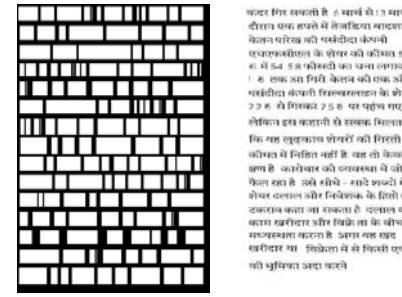


Figure 3. Word Flow

3.2. Word Model Based Compression

We have developed a symbolic representation for the word images [14] of Devnagari based scripts. For this purpose,word images are represented in the form of a Geometric Feature Graph(GFG).GFG is a graph based representation of the features extracted from the word image. The GFG used has the following specifications [14].

- The nodes in the GFG can be either the end points or the junction points.

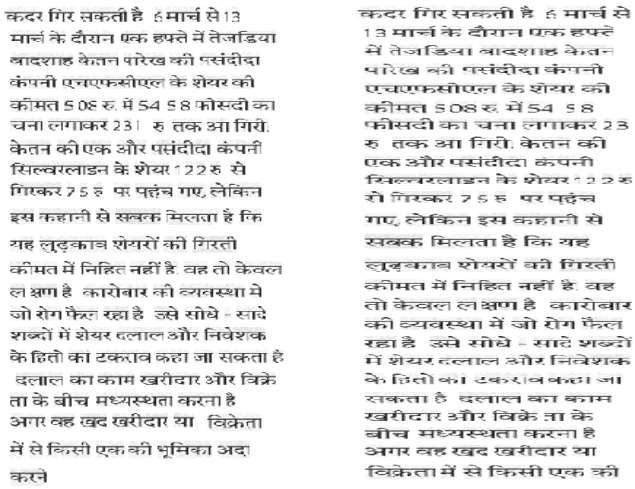


Figure 4. Word Flow for different screen sizes

- The branch between any two nodes represents the type of the generic shape connecting the two node points in the word image.

Different type of connectors are shown in fig 5

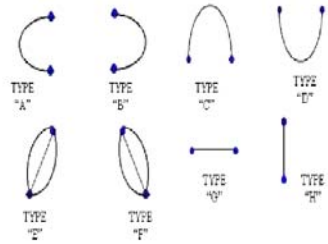


Figure 5. Different type of connectors

A skeletal representation of segmented word image is obtained by a thining algorithm.The *'skeleton obtained'* may have extra pixels and breaks. These errors are eliminated using a set of smoothing heuristics. The GFG is constructed by following this skeleton of the thinned image of a word.The skeleton tracing algorithm traces the connected paths in the words in a specific order of direction and identifies terminal points and multiple connected pixels as potential nodes of the GFG.So, after allocating nodes, they are ordered according to their x-y coordinates making node position independent of traversal path. Next, branches of GFG are labelled using shape of the connectors. Detailed algorithm for GFG extraction is presented in [14].

Using GFG model of the word, in this paper, we have proposed a novel feature based compression scheme. The

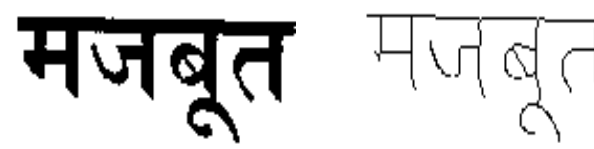


Figure 6. Example of skeleton of word images obtained

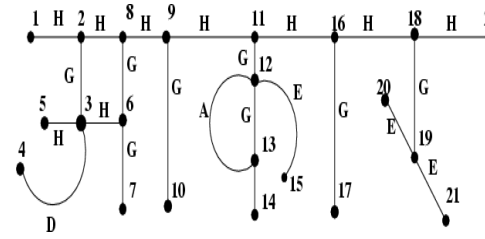


Figure 7. GFG of Hindi Word Akar

compression scheme consists of replacement of the word image by a string of symbols obtained from the GFG. We can encode the GFG as a string by traversing the word GFG in DFS(depth first Search) fashion. The branches of the GFG are labeled and an unambiguous ordering of the nodes is achieved. Branch labeling is done using the terminal points xt_1, yt_1, xt_2, yt_2 of the connected nodes and the $x_{min}, y_{min}, x_{max}, y_{max}$ from the path traversed from the one node to other. Let xt_1, yt_1, xt_2, yt_2 are the coordinates of the two connected nodes. Let x_{min}, y_{min}, x_{max}, y_{max} be the minimum and maximum values of the coordinates obtained while travelling from node1 to node2. Now,

$$x_{min-diff} = min(\mid xt_1 - x_{min} \mid, \mid xt_2 - x_{min} \mid)$$

$$x_{max-diff} = min(\mid xt_1 - x_{max} \mid, \mid xt_2 - x_{max} \mid)$$

$$y_{min-diff} = min(\mid yt_1 - y_{min} \mid, \mid yt_2 - y_{min} \mid)$$

$$y_{max-diff} = min(\mid yt_1 - y_{max} \mid, \mid yt_2 - y_{max} \mid)$$

A set of rules based on these parameters is used to identify generic nature of the connectors. Exploiting the (x,y) coordinates associated with the nodes we have fixed the following order for visiting children at a node:down, left, right and top.Consequently,looking at the symbols in the string we can place them unambiguously in correct spatial layout with respect to its parent without using (x,y) value of the nodes at the receiver. We also add a back tracking symbol $ at the backtrack point for indicating order in which nodes need to be considered.In addition we store length of the primitive in terms of the number of pixels in the source image.Algorithm for encoding is given in(Figure 8).We further

ALGORITHM ENCODING

```
A.REPEAT FOR ALL branches.
B.N1 is Node1,N2 is Node2
C.B1 is Branch1
1.Maintain N1 and N2 of each branch
in GFG.
2.Visit the first branch of GFG,
  set current pointer to N2 of B1.
3.Make this branch as visited branch.
  Add branch type of B1 in GFG string.
4.Check for new branch having N1=N2
  of previously traced branch.
5.Check the unvisited branches.
6.Trace the next unvisited branch in
  the order as down, left, right, top.
7.Make this branch as visited branch.
  Add branch type to encoded string
  and also add length of primitive.
8.Move current pointer to the N2
  of visited branch.
9.IF N2 of visited branch is the end,
  node THEN add $ symbol in the encoded
  string and move pointer to position
  of the previous node.
```

Figure 8. Algorithm for Encoding

use Huffman coding for the symbols in a textual component for compression of the data. Effectively, we have derived a symbolic compression scheme for the textual component of the document image. However, this scheme does not guarrante reproduction of the textual image using the same font. This compression scheme, although is conceptually similar to the technqiues suggested as part of JBIG2, has no commonality with processing and representation methodology followed in JBIG2 [15]. For example the encoded GFG of the hindi word AKAAR (see figure 7) is
H10G5D5$H5$H5G7$G7H7$H7G7B8H8G5
G5$A5E7$$$$H5G$H6GE6$E6$$H10.
This encoded string corresponding to the text image is send from server to client.

Decoding process involves graphics based regenaration of the word image using the encoded string. The first node is assigned a logical coordinate on the target display device. All primitives are drawn with respect to the position of the parent node in the correct sequence. In case of backtrack symbol current pointer is moved to that of the parent node. Algorithm for decoding is given in (figure 9). For example, after decoding regenerated image word AKAAR is shown in figure 10. We can see from the fig 10 that we can have regenerated word image of varying size since coded string

ALGORITHM DECODING

```
A.REPEAT FOR each character in the
  string
1.Identify the drawing position
2.Maintain a start and end node of
  each primitive.
3.Maintain a current pointer which
  indicates the position from where
  the next element will be starting
4.Start with first primitive,draw
  graphically the corresponding
   primitive,
  size of which in terms of diameter
  or length is specified in the
  encoded string itself.
5.Move the current pointer to the
  end node of drawn primitive.
6.Whenever a $ symbol is seen,move
  current pointer to the position of
  previous node in the stack.
7.Repeat the steps,until each GFG
  string character is processed.
```

Figure 9. Algorithm for Decoding

contains size information about the primitives in the original word image. While decoding we are maintaining a relative scale factor.Whenever user specifies the font size, original size of word image is multiplied by the factor specified by the user. Here, we have flexibility of varying size of word images. Depending on the size of the display we can even reduce the size of the word.

Figure 10. Example of varying size of decoded image words

For establishing efficacy of the compression scheme we have compared our compression scheme with JBIG1 [8]. We have taken uncompressed .pgm files of images having only textual component and applied our compression scheme and JBIG1 compression. Some of the comparison results are presented in table 1.Our compression scheme drastically outperforms JBIG1. However, our scheme is applicable only for Devanagari script based documents be-

cause of the nature of the graphical primitives used in defining GFG. Further, correctness of reconstruction depends on the quality of the input data and feature extraction scheme.

Pgm file	Jbig(compression)	encoded file
bytes	bytes	bytes
206,272	4,894	1,290
135,994	1,064	120
11,619	892	647
220,347	1,469	493
407,721	12,592	912
428,457	13,275	2,515
163,725	5,111	350
220,346	4,820	319

Table 1. Comparison between Jbig and encoded file after Huffman coding

4. Image Transcoding

In this section we have explained, how the transcoding of image part of the document image is done. For transcoding image part we are following saliency based view extraction scheme proposed in [9]. For large images, we have developed an optimal path algorithm for sequential presentation. For contrast based saliency mapping [7] we have considered an MxN image as a perceive field with MN perception units,if each prception unit contains one pixel.The contrast value calculated is normalized to [0,255],which forms the saliency map.Using the contrast information attention centre and attended view is obtained. A fuzzy region [7] growing algorithm is used to locate image regions of interest.Example of saliency based region of interest extracted is shown in fig 11. An optimal image browsing path [4] is then calculated based on the image attention model.Example of optimal browsing path is shown in fig 13

Figure 11. Examples of Saliency View Extraction

Figure 12. Original Image

Figure 13. Example of Sequence of Images as Browsing Path

5. Summary Generation and Document Presentation

Automatic generation of summary of a document represented in terms of images is a challenging problem. We have used conceptual model of the documents encoded in ontology representation language DAML/OIL for summary generation. Ontology encodes different conceptual entities and their relations expected in a class of documents. For example, for the class of newspapers, ontology specifies existence of front-page, editorial page, sports page and other pages as document components. For each of these components, we have the DTD model which depicts visual layout of the page. During model guided segmentation,[5],[6], the page class and components are identified. Textual components are also ordered according to their font size. We use these labels of segments to generate domain specific summaries. The summary generation scheme is encoded as a set of rules and represented using XML. For example, for a newspaper, summary consists of name and logo of the newspaper, textual component of large font size from front page, sports page and editorial page and transcoded images from front page and sports page.An example of document image summarization is given in fig14. There exist a browsing path for each of different components of a document for example, front page news, sports news and editorial news.The user can choose any one of the options,based on which,all the headings that fall into that category are displayed on screen. On this page again, we have option to see details of headings in the same category. If we choose any one of the options for headings,then the text in context with that heading is displayed on the screen.There is also an option to see the images associated with the same text.In this way, browsing based on tree structure is provided at client device

Figure 14. Requesting for Document Images: *In the fig,various segments of a newspaper have been shown.As per user's choice,he can request for titles,headlines,images concerned with headlines,image sequences and a description corresponding to the headline.*

for downloading components only on demand.

6. Conclusion and future scope

This paper presents a scheme for transcoding Devnagari script based Indian language documents. It presents a novel symbolic compression scheme for textual component of document images and an ontology based document summary generation scheme.

References

[1] M. A.Smith and T. Kanede. Video skimming for quick browsing and image characterization. Technical Report TR # 186, School of Computer Science, Carnegie,Mellon University, 1995.

[2] A.Srinivasan, K. Das, and S.Chaudhury. Preprocessing Algorithms for Understanding Images of Indian Languages documents. In *International conference on Pattern Recognition,Image Procesing and Computer Vision(ICPIC)*, pages 143 – 148, Dec 13-15-1995.

[3] F. chen and D.Bloomberg. Summarization of imaged documents without OCR. In *Computer vision and image understanding*, June 1998.

[4] L. D. *Contrast Sensitivity.Chapter 5.Im:Cronly-Dillon:Vision and Visual dysfunction*. London:Macmillan Press., 1991.

[5] G.Harit, S.Chaudhury, and H.Ghosh. Managing Document Images in a Digital Library:An Ontology guided Approach. In *International workshop on Document Analysis for Digital Library-2004,Palo Alto Research Center,CA 94304 USA*, pages 64 – 92, 2004.

[6] G.Harit, S.Chaudhury, P.Gupta, and N.Vohra. A model guided document image analysis scheme . In *Sixth International Conference on Document Analysis and Recognition,10-13 Sept.2001*, pages 1137 – 1141, 2001.

[7] G.J.Klir and B.Yuan. *Fuzzy sets and fuzzy logic:Theory and Applications*. Prentice Hall,Upper Saddle River,NJ, 1995.

[8] M. Kuhn. *The latest release of JBIG-KIT is http://www.cl.cam.ac.uk/ mgk25/download/jbigkit-1.5.tar.gz*. 2003.

[9] L.Chen, X.Xie, X.Fan, W.Y.M, H.J.Zhang, and H.Q.Zhou. A visual attention model for adapting images on small displays . *ACM Multimedia Systems Journal, Springer-Verlag*, Vol.9(4):353 – 364, 2003.

[10] T. M.Breuel, W. C.Janssen, K. Popat, , and H. S.Baird. Paper to PDA. In *IAPR 16th ICPR,quebec City,Canada*, volume 4, pages 476 – 479, August 2002.

[11] K. Nagao and K. Hasida. Automatic text summarization based on the Global Document Annotation. . In *COLING-ACL'98*, 1998.

[12] K. Nagao, S. Hosoya, Y. Shirai, and K. Squire. Semantic transcoding: Making the world wide web more understandable and usable with external annotations. In *COLING'2000 Workshop on Semantic Annotation and Intelligent Contents*, 2000.

[13] R.Sproat, P.Taylor, M.Tanenblatt, and A.Isard. The SABLE Consortium.A Speech Synthesis Markup Language.http://www.cstr.ed.ac.uk/projects/ssml.html. In *EUROSPEECH 97,Rhodes,Greece*, 1997.

[14] S.Chaudhury, G. Sethi, A.Vyas, and G. Harit. Devising Interactive Access Techniques for Indian Language Document Images . In *ICDAR*, pages 885 – 889, 2003.

[15] Y.F.Ma and H.J.Zhang. Contrast based Image Attention Analysis Using Fuzzy Growing. In *International Multimedia Conference archive Proceedings of the eleventh ACM international conference on Multimedia table of contents Berkeley, CA, USA SESSION:Image annotation and video summarization table of contents*, pages 374 – 381, 2003.